SECURE COVERT COMMUNICATIONS OVER STREAMING MEDIA USING DYNAMIC STEGANOGRAPHY

JINGHUI PENG

A thesis submitted in partial fulfilment of the requirements of The University of West London for the degree of Doctor of Philosophy

February 2020

TO MY FAMILY AND MY FRIENDS

ABSTRACT

Streaming technologies such as VoIP are widely embedded into commercial and industrial applications, so it is imperative to address data security issues before the problems get really serious. This thesis describes a theoretical and experimental investigation of secure covert communications over streaming media using dynamic steganography. A covert VoIP communications system was developed in C++ to enable the implementation of the work being carried out.

A new information theoretical model of secure covert communications over streaming media was constructed to depict the security scenarios in streaming media-based steganographic systems with passive attacks. The model involves a stochastic process that models an information source for covert VoIP communications and the theory of hypothesis testing that analyses the adversary's detection performance.

The potential of hardware-based true random key generation and chaotic interval selection for innovative applications in covert VoIP communications was explored. Using the read time stamp counter of CPU as an entropy source was designed to generate true random numbers as secret keys for streaming media steganography. A novel interval selection algorithm was devised to choose randomly data embedding locations in VoIP streams using random sequences generated from achaotic process.

A dynamic key updating and transmission based steganographic algorithm that includes a one-way cryptographical accumulator integrated into dynamic key

exchange for covert VoIP communications, was devised to provide secure key exchange for covert communications over streaming media. The discrete logarithm problem in mathematics and steganalysis using t-test revealed the algorithm has the advantage of being the most solid method of key distribution over a public channel.

The effectiveness of the new steganographic algorithm for covert communications over streaming media was examined by means of security analysis, steganalysis using non parameter Mann-Whitney-Wilcoxon statistical testing, and performance and robustness measurements. The algorithm achieved the average data embedding rate of 800 bps, comparable to other related algorithms. The results indicated that the algorithm has no or little impact on real-time VoIP communications in terms of speech quality (< 5% change in PESQ with hidden data), signal distortion (6% change in SNR after steganography) and imperceptibility, and it is more secure and effective in addressing the security problems than other related algorithms.

Keywords: Covert communications, hardware random key, key distribution, steganography, VoIP

ACKNOWLEDGEMENTS

I would like to thank my principal supervisor, Professor Shanyu Tang, PhD, FBCS, for his guidance, advice and support over this work. His enthusiasm and engagement in my area of research, *i.e.* Information Security, kept me motivated in every stage of my PhD journey.

I wish to thank my supervisor, Professor Graham Brooks, PhD, for his guidance and advice in some of the theoretical and ethical aspects of this work.

I am grateful to my supervisor Dr Anastasia Sofroniou for her advice and suggestions in some of the practical aspects of this work.

Special thanks to the Graduate School staff, especially Maria Pennells and Kiranjit Johal, for their support and encouragement throughout my PhD study.

I would like to thank various industrial sponsors for their financial supports in the form of an international PhD scholarship.

RELATED PUBLICATIONS

Peer-reviewed Journal Papers:

J. Peng, S. Tang*, “Covert communication over VoIP streaming media with dynamic key distribution and authentication,” *IEEE Transactions on Industrial Electronics*, in Press, 26 Feb. 2020. (doi: 10.1109/TIE.2020.2979567) (impact factor: 7.503)

J. Peng, S. Tang*, J. Li, “Fast fourier transform-based steganalysis of covert communications over streaming media,” *International Journal of Computer and Information Engineering*, vol. 13, no. 7, pp. 362-367, Jul. 2019. (doi: 10.5281/zenodo.3299971)

Manuscript Submitted to Journals:

J. Peng, S. Tang*, “Real-time steganographic system with embedded VoIP for secure communications,” submitted to *IEEE Transactions on Industrial Informatics*.

Conference Presentations:

J. Peng, S. Tang, J. Li, “FFT-based steganalysis of covert communications over streaming media,” in Conference Proceedings of ICCSE 2019: 21th International Conference on Computer Science and Engineering, Paris, France, 18 July 2019, pp.734-739. (Won “Best Paper Award”)

J. Peng, “Self-adaptive steganographic scheme with chaotic map for secure covert VoIP communications,” in Proc. UWL Doctorial Students’ Conference 2019, London, UK, 29 May 2019, pp. 15.

J. Peng, S. Tang, “Steganography with AES encryption for secure real-time VoIP covert communications,” presented at Annual Research Showcase, University of West London, UK, 29 January 2019.

CONTENTS

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1 Introduction

This chapter first gives a brief overview of what the PhD project involves. It then describes the main motivation for carrying out the research project, research problems to be addressed, and a significant contribution the research has made to the scientific knowledge in the related fields of covert communications and steganography. Finally, the chapter covers the outline of the thesis, providing a general description of successive chapters.

1.1 Overview

According to the authorities, the main threats to UK national security are terrorism, espionage, cyber threats, and other threats such as the proliferation of weapons of mass destruction and organised crime. Among them, cyber threats have immediate impact on the UK's national information and communications technology (ICT) infrastructure.

The expansion of the Internet provides countless opportunities for cyber crime to be committed. Cyber crime occurs in a networked environment such as the Internet, Cloud, the Internet of Things (IoT), etc. It includes the following criminal activities: Computer hacking and cracking; Malware and automated computer attacks: developing and / or spreading malicious code (e.g. virus and Trojans); Digital piracy,

online fraud, intellectual property theft; Spamming (sending unwanted or junk e-mails); Network intrusion; Networked based or network enabled crime such as phishing, identity theft, IPR crime, distribution of child pornography; Online extremism, cyber terror, and cyber warfare. The challenges arising from the growth of cyber crime need to be dealt with.

The UK economy largely depends on its ICT infrastructure to digitally support businesses, commerce and private citizens. As the Internet permeates all levels of society, hostile actors are seeking to take advantage of people's increasing Internet dependency to launch cyber terrorism, state or industrial cyber espionage, and cyber crime. Hence, advanced security countermeasures and technologies are urgently sought to counter these cyber threats.

Information security and privacy technologies, such as encryption, digital signatures, authentication, enterprise key management, and biometric identification systems, are based on classical cryptography. However, these cryptographic technologies are being challenged by increasingly sophisticated attack tactics due to a sharp increase in computing power. It is generally believed that one promising solution to the data protection problem is digital steganography, which is the science of concealing a message or information within other non-secret text or data.

Digital steganography complementing advanced cryptography, such as true random key generation, advanced encryption standard, and dynamic key distribution and management, is expected to lead the way in counteracting the latest cyber attacks, including quantum adversaries.

1.2 Motivation and Research Problem

Data security is of great interest to businesses and governments alike and has received increasing attention in recent years. Finding a solution to the data protection problem is one of the greatest challenges faced by cyber security researchers today.

As the Internet permeates every level of the infrastructure of a country, society and organisation, cyber attacks are increasing at an alarming rate. The ICT infrastructure has been a great support to the industries that are information intensive in one way or the other, which are considered one of the most important economic sectors for a variety of reasons.

The ICT dependent industries include financial services (insurance industry), creative (advertising, fashion, film, music, video games), education, health care, hospitality, professional services, software, and tourism industries, as well as some manufacturing and agriculture industries.

Financial services encompass a wide range of businesses that manage money, including credit unions, banks, credit-card companies, insurance companies, accountancy companies, consumer-finance companies, stock brokerages, investment funds, individual managers, and some government-sponsored enterprises. So it matters to individuals, organisations and governments.

Among these industries, the financial services industry is under real threat from an exponential rise in the scale and significance of cybercriminal capability. Hence, cutting-edge technologies are urgently sought by academics and engineers to protect

data against advanced social engineering and evolving sophisticated attack tactics.

To date, modern cryptography is widely used for data protection in a pervasive computing environment, such as networked systems and the Internet. Classical security measures are built on encryption, which is the activity of converting data or information into code in such a way that only the intended recipient can determine its meaning.

Whilst encryption algorithms are based on computational hardness assumptions, encryption is facing challenges due to a sharp increase in computing power that has led to decryption of several encryption algorithms, indicating possible vulnerabilities in the encryption primitives. It is generally recognised that a major drawback to encryption is that the existence of encrypted data is not hidden.

As a revival, quantum cryptography can detect attempts at hacking but the communication range only reaches about 93 miles, far short of the distance requirements needed to transmit information with modern computer and telecommunication systems. Moreover, it is not yet clear whether there is a commercial market for this extremely expensive technology.

Digital steganography aims to hide confidential data well so that unintended recipients would not even notice the existence of the hidden confidential data. It is otherwise regarded as a promising technology to complement cryptography in addressing data protection issues.

The Internet enables Voice over Internet Protocol (VoIP) to provide reliable, global, low-cost and/or even free services, so many users communicate with each other daily

using VoIP products, leading to increasing traffic of VoIP streams transmitted over the Internet. As an interesting subject in the field of information security, steganography or covert communication works by hiding messages in inconspicuous cover objects (e.g. VoIP streams) that are then sent to the intended recipient. Steganography can provide an additional layer of security in addition to encryption by embedding the encrypted message into steganographic carriers, which helps individuals or organisations protect sensitive information. It can be used in academic, commercial and military applications. For example, a message can be steganographically embedded into the least significant bits of frames on a CD. Covert steganographic channels can be used to bypass the censorship in a hostile environment. The covert channel can also be used by the adversary as a possible means of information exchange. A message can be concealed before distribution by splicing it to the end of a copy of a normal audio or video. A disgruntled employee may use steganography to ship out the most commercially sensitive information.

1.3 Research Questions

Covert VoIP communication is based on steganography and cryptography. The basic process of a covert VoIP communication scenario consists of two phases: signalling phase and conversation phase. The signalling phase sets up and negotiates VoIP session parameters between the communicating parties. Mutual authentication is used to convince parties of each other's identity and to exchange session keys. In the conversation phase, the sender embeds encrypted secret data into covert objects and sends it to the receiver. After receiving and extraction, the receiver decrypts the

massage using the same key and obtains the secret message.

In the process of covert VoIP communications, several issues need to be addressed to provide security, including a model for covert steganographic communication over streaming media, secure keys, data embedding interval selection, and secure dynamic key updating and transmission. Chapters 4 - 6 constructed a theoretical model and devised algorithms for covert VoIP communication systems to solve these problems correspondingly.

Early research in digital steganography focused on image steganography, which covered a variety of topics that are well summarised by Cox et al. (Cox et al., 2008) and Ker (Ker, 2014). Since the era of evolving network applications and mobile communications (3G/4G/5G), research has shifted to steganography in streaming media, like Voice over Internet Protocol (VoIP), which acts as a resilient covert communications channel to protect data transmitted over the Internet and in a cloud environment.

The current literature in streaming media steganography is somehow extensive. Related work has been devoted to devising steganographic algorithms while the security offered by number-based secret keys has been sidestepped due to technical complexity. In reality, the successfulness of steganographic algorithms used for data protection relies largely on the secret keys and the transmission of keys between the communicating parties. Unfortunately, the secret key generated by a pseudorandom-number generator is not that secure, and given enough time and computational power, the key would be unencrypted by malicious attackers.

Security in transmission of secret keys is more crucial for streaming media steganography due to the timing and loss of packets, *i.e.* it requires continuous embedding and the necessary synchronisation of sender and receiver. Thus far, there has been no proven secure key transmission method that could be put into use for streaming media steganography. Hence, this research is intended to fill the gap in this area of research and increase knowledge as it addresses this challenging and important problem.

As quantum technologies have advanced in recent years, existing steganographic / cryptographic techniques have had to be modified in order to cope with quantum adversaries. This research is expected to make a major technological breakthrough in hardware-based true random key generation and dynamic key updating and transmission, which would combat quantum adversaries. The research attempts at revealing the potential of true random key generation and dynamic key updating and transmission for innovative applications in the field of covert VoIP communications using streaming media steganography.

There are some key problems associated with covert VoIP communications, such as lack of a practical information theoretical model, true random key generation, and secure dynamic key updating and transmission, in an attempt to solve the uncertainty of the data embedding rate of streaming media packets and ensure the integrity of the secret message to be hidden using streaming media steganography.

This project addresses the following research questions:

A. **How can an information theoretical model of secure covert communication**

depict the security scenarios in streaming media-based steganographic systems?

Previous modelling work has considered private key steganography or public key steganography, with passive or active attacks, in static cover objects such as texts, images and audio files. However, existing information theoretical models for steganography in static cover objects cannot be put into use for steganography in streaming media (like VoIP) because of the required continuous data embedding process and the necessary synchronization of sender and receiver due to packet loss in communications over streaming media. Hence, a new information theoretical model of secure covert communications over streaming media is required to take into account the characteristics of communications over streaming media like VoIP.

An information source might be modelled as a stochastic process, but it is complicated and time consumable in practice. A stochastic process can be used to model an information source for covert VoIP communications using streaming media steganography, to decide whether the resultant probability distributions of cover objects (*i.e.* streaming media) meet the requirements of the theoretical model.

B. **How does hardware-based technology generate true random numbers as secret keys for covert VoIP communication using digital steganography to ensure the security of cryptographic systems?**

Cryptographical keys play a fundamental role in using steganography and cryptography to achieve a covert communications channel to protect data from sophisticated cyber attacks. Presently, cryptographical keys used for covert

communications over streaming media are pseudorandom numbers generated by software that uses a predictable process to yield pseudorandom numbers. Pseudorandom keys are subject to compromise due to a sharp increase in computational power, which means the data encrypted with the pseudorandom keys is not secure.

A true random number generator or hardware random number generator is a device that generates true random numbers from a physical process, rather than by means of an algorithm. Secret keys generated by a true random number generator play an important role in ensuring the confidentiality of the data to be hidden using digital steganography during covert communications over streaming media.

C. **How does a one-way cryptographic accumulator work along with dynamic key updating and transmission, which is integrated with the data embedding and extraction processes in covert VoIP communications?**

As steganography underpinning covert communications over streaming media require the secure transmission of a series of secret keys, this project needs to provide a fundamental understanding of how dynamic key updating and transmission can work along with data encryption, data embedding and data extraction (parts of digital steganography) in covert communications over streaming media like VoIP, in a timely manner to cope with cyber attacks and potential quantum adversaries.

There is currently no proven secure key transmission method that could be put into use for streaming media steganography. To advance knowledge, this research proposes a novel dynamic key updating and transmission based steganographic

algorithm, consisting of a one-way cryptographic accumulator, for covert communications over streaming media.

***D*. How can a statistical test be used to analyse the adversary's detection performance?**

Non parameter Mann-Whitney-Wilcoxon statistical testing and T-test can be used to analyse the adversary's detection performance on distinguishing between an innocent cover object and a modified stego object containing a hidden message.

1.4 Research Aim

There are some key problems associated with covert VoIP communications, such as the theoretical model, the uncertainty problem of the embedding rate of media packets and the integrity of the secret message.

The aim of this project is to examine the characteristics of time-variance of data payloads in the process of covert communications using digital steganography in streaming media, with a special emphasis on constructing an information theoretical model for covert communications over streaming media like VoIP. Through devising a secure dynamic key updating and transmission protocol based on a one-way cryptographic accumulator and a series of secret keys that are generated by a hardware based true random number generator, applicable to covert communications over streaming media, the project intends to explore a new approach to improving the integrity, secrecy and robustness of the secret message being hidden, thereby addressing the non-integrity problem of the secret message to be protected.

Specifically, the objectives of this research include:

A. Develop an information theoretical model for covert steganographic communication over streaming media;

B. Devise a new algorithm that uses hardware as the entropy source to generate true random keys as dynamic keys to be used by covert VoIP communications;

C. Select data embedding locations in real-time media streams for covert VoIP communications;

D. Devise a secure dynamic key agreement and updating algorithm based on One-way accumulation, which is applicable to covert VoIP communications.

1.5 Research Methodology

The research methodology includes using an information-theoretic model to guide devising a steganographic algorithm, building a covert communications platform, conducting performance testing, and carrying out security evaluation on the proposed algorithm. The full details of the methodology used in this research are as follows:

A. Collect and analyse a large amount of experimental data from in-house covert VoIP communications experiments.

B. Construct a new information theoretical model for secure covert communications over streaming media, depicting the security scenarios in streaming media-based steganographic systems with passive attacks.

The model includes three components: a stochastic process that models an

information source for covert VoIP communications using streaming media steganography, the theory of hypothesis testing that analyses the adversary's detection performance, and a discrete prediction model of high precision that simulates the characteristics of time-variance of streaming data payloads.

C. Investigate ways in which hardware entropy sources can be used to generate true random numbers as secret keys for covert communications using streaming media steganography, to ensure the data they protect remains absolutely secure.

D. Devise a dynamic key updating and transmission based steganographic algorithm that includes a one-way cryptographical accumulator integrated with dynamic key exchange for covert communications over streaming media. The new algorithm can protect data from cyber attacks, such as the man-in-the-middle attacks, which threaten almost all existing steganographic algorithms.

E. Examine how the proposed true random numbers and dynamic key distribution based steganographic algorithm can work with covert communications over streaming media.

F. Devise a true random keys and dynamic key updating and transmission based steganographic algorithm for covert communications over streaming media.

G. Accomplish security analysis of covert communications using dynamic steganography in streaming media, by means of steganalysis using non parameter statistical analysis.

H. Carry out performance and security measurements and evaluation on covert communications over streaming media, studying the factors that mostly affect covert

VoIP communications that are underpinned by dynamic steganography in streaming media.

1.6 Contributions

The emergence of a pervasive computing environment, such as networked systems and the Internet, poses new challenges for the security of information systems, which are widely used at every level of the UK society.

This research provides a novel technological approach to data protection for networks and IT systems, which would contribute to the development of the cyber security sector and digital creative industries around the world, in an attempt to solve real security problems in the financial, banking, and creative industrial sectors.

The research has several contributions to the field of streaming media steganography which can be summarised as follows:

A. Due to the required continuous data embedding process and the necessary synchronisation of sender and receiver in streaming media communications, existing information theoretical models of steganography in static cover objects cannot be put into use for streaming media steganography underpinning covert VoIP communications.

Research into developing an information theoretical model for covert communications using steganography in streaming media is still in its infancy. To meet the requirements, this research constructs a new information theoretical model of secure covert communications over streaming media by taking into account the

characteristics of streaming media like VoIP.

B. Most cryptographical keys are generated by software that uses a predictable process to yield pseudorandom numbers. Pseudorandom keys are subject to compromise, which means data encrypted with such keys is not secure. So far, no literature has been found on the use of true random keys to enhance the security of streaming media steganography for covert communications, and a novel feature of this research is the utilisation of hardware as an entropy source to generate true random keys, in order to secure streaming media steganography underpinning covert communications over streaming media.

C. There is currently no proven secure key transmission scheme that could be put into use for covert VoIP communications using streaming media steganography. To advance knowledge, this research devises a novel dynamic key updating and transmission based steganographic algorithm for covert communications over streaming media, which involves a specially designed one-way cryptographic accumulator and dynamic key exchange integrating with the data embedding and extraction processes.

D. Two minor contributions of this research are the use of a stochastic process to model an information source for covert VoIP communications using streaming media steganography, and the use of the theory of hypothesis testing to analyse the adversary's detection performance on distinguishing between an innocent cover object and a modified stego object containing a hidden message.

E. The last contribution of this research is its methodology. It presents a new

steganographic approach that seamlessly integrates advanced encryption standard, true random key generation, and dynamic key updating and transmission with the data embedding and extraction processes, thereby realising secure covert communications over streaming media and achieving a great data embedding capacity comparable to other related steganographic algorithms.

1.7 Thesis Outline

Chapter 1 of this thesis provides introductory material, consisting of a brief overview of the research project, the main motivation for carrying out the project, research problems being addressed, and a significant contribution the research is made to scientific knowledge in the field of digital steganography.

Chapter 2 includes an in-depth discussion of how VoIP communications work and security issues that may occur in a VoIP communications system. It involves components of VoIP (end-user equipment, network components, call processor, gateways, and protocols), limitations of the best-effort IP service, i.e. packet loss, end-to-end delay, and packet jitter, and protocols for real-time conversational applications, such as Real-Time Transport Protocol (RTP) and Session Initiation Protocol (SIP). It then presents security analysis of VoIP communications, identifying a few types of malicious attacks to which VoIP communications might be subjected, along with possible countermeasures.

Chapter 3 describes a number of computer and network security concepts related to designing steganographic systems, examines the basic hardware and software

components that make up a steganographic system, and discusses classifications of steganographic methods, properties and evaluation criteria of steganography such as undetectability, imperceptibility, security, capacity and robustness. The chapter covers various attacks on steganographic communications and provides a summary and synthesis of published information on steganography in streaming media.

Chapter 4 focuses on covert communications over VoIP streaming media. After introducing some basic terminology and concepts, the chapter discusses Cachin's definition of steganographic security to identify the limitations of its usage in streaming media steganography. It then describes a new framework for modelling secure covert VoIP communications, and of using this information theoretic model in a steganography underpinning covert VoIP communications system. It also examines AES-128, data embedding and data extraction algorithms that are applicable to covert VoIP communications. Covert VoIP communications experimental set-up and evaluation criteria, e.g. the steganographic bandwidth, undetectability, the packet loss rate, the perceptual evaluation of speech quality value, and the signal-to-noise ratio, are discussed in the second half of the chapter.

Chapter 5 explores the potential of hardware based true random key generation and chaotic logistic map for innovative applications in covert VoIP communications over streaming media. It then investigates ways in which hardware entropy sources can be used to generate true random numbers as secret keys for streaming media steganography underpinning covert communications, to ensure the data they protect remains absolutely secure.

The chapter also describes, tests, and analyses the true random keys based

steganographic algorithm that underpins covert VoIP communications, comparing the steganographic bandwidth, undetectability, the packet loss rate, the perceptual evaluation of speech quality value, and the signal-to-noise ratio, etc., which are measured using a Digital Speech Level Analyser. Security analysis using non parameter M-W-W statistical test is utilised to prove the security of the new algorithm.

Chapter 6 covers novel techniques for using one-way cryptographical accumulators to realise dynamic key updating and transmission for covert VoIP communications. This includes the areas of key distribution and management and security analysis using t-test, which become complicated when streaming media are acted as cover objects for steganography. It then examines a dynamic key updating and transmission based steganographic algorithm that can protect data from cyber attacks, such as the man-in-the-middle attacks, which threaten almost all existing steganographic algorithms.

This chapter also demonstrates the effectiveness of the proposed steganographic algorithm, discussing the effect of increased complexity of the algorithm, at variable embedding interval distances, on performance and security. It then examines the algorithm by varying the size of the data to be hidden and protected, to determine the robustness of the algorithm in terms of data embedding capacity.

Chapter 7 ‘Conclusions and Future Perspectives’ begins at the overview of the project and summarises research findings and innovations. It then looks at research limitations in the design of the study, such as difficulty in designing an effective mechanism for resisting tampering attacks. Finally, it points out the potential directions for future research.

CHAPTER 2 VoIP Communications and Security

Streaming media communication such as Voice over Internet Protocol (VoIP) is one of the most popular real-time services on the Internet. This chapter discusses how VoIP works and the security issues that may occur in a VoIP communication system. After introducing the background, features and components of VoIP, it presents the principles of VoIP communication. Then, it describes the limitations of the best-effort IP service and the key technologies to deal with a variety of problems of VoIP. This chapter also covers the security analysis of VoIP communications and identifies some types of security issues and possible countermeasures.

2.1 Introduction to VoIP

VoIP is a digital transmission technology which provides a real-time voice communication service on the basis of the Internet. It is a telephone application implemented on the Internet Protocol (IP) network through the Transmission Control Protocol (TCP) / IP protocol. Therefore, VoIP is also called IP phone or network phone.

VoIP can be achieved on any networks based on an internet protocol, such as the Internet, Intranet and Local Area Networks (LANs). VoIP applications include Personal Computer (PC) to PC connection, PC to Public Switched Telephone

Network (PSTN) or PSTN to PC connection and PSTN to PSTN connection (Nagirireeddi, 2008). Its main services include voice services and real-time fax services over IP-based networks, interactive voice response (IVR) services implemented on the Web, and a variety of communication services such as E-mail and real-time telephone. VoIP services operate on an Internet protocol to transmit compressed voice samples as frames and messages as a group of bytes over an IP data network (Nagirireeddi, 2008). In VoIP applications, voice from end-user equipment is converted into a signal level, digitised, compressed as voice payload, and sent as IP packets.

The public switched telephone network is mainly based on circuit switching, and the cost is high. With the development of network and the advancement of voice coding technology, voice communication is gradually developing to low-cost integrated communication based on IP packet switching. VoIP has achieved breakthrough progress and substantial application. This kind of voice communication on the network improves the quality and reliability of voice transmission, promotes the utilisation of network resources, and reduces the cost of voice communication. Therefore, VoIP has become one of the fastest growing and most popular technologies in the world.

Compared with traditional voice communication, VoIP technology has many unique advantages. Features such as low cost, extensibility, full use of network resources and in line with the development trend of the integration of three networks (Telecommunications network, Broadcast network and the Internet), make VoIP gradually occupy the traditional telephone business market and become one of the

most popular technologies in the computer science field.

A. Low cost

Low cost is one of the main advantages of VoIP. The VoIP system connects PSTN and the Internet mainly through an IP telephony gateway, thus realising three kinds of calls between computer and telephone, i.e., PC to PC, PC to PSTN or PSTN to PC and PSTN to PSTN (Nagirireeddi, 2008). Since the VoIP network is organised by devices such as gateways and connected to users by PSTN, a low-cost IP network is used instead of an expensive long-distance transportation network mainly based on ‘copper core’, thereby greatly reducing communication costs.

The VoIP system also minimises the cost of business communications for enterprises. Industry sectors and large enterprises (such as government, bank, etc.) mostly have dedicated data communication networks and telephone communication networks due to information constriction. The data communication network usually needs to lease DDN special line etc., and the telephone communication network usually needs to build telephone switches and rent relay stacks of telecommunications. Not only do they have to pay for the rental of the line, but they also have to bear higher costs for use and maintenance. The development of information technology has made it possible for these industry users and large enterprises to build a dedicated IP broadband network for the integrated transmission of voice, data and image services, thereby reducing their communication costs.

B. Extensibility

VoIP adopts an open transport protocol architecture, which facilitates connectivity and

standardisation among vendors' products. The IP communication network can support the transmission of multimedia information such as voice, image and data at the same time, which is conducive to the integration of multimedia services (Nagirireeddi, 2008). In addition, for industry users and large enterprises, VoIP systems have easy-to-add features and extensibility, which simplifies the communication management of enterprises.

C. Full use of network resources

The traditional telephone network PSTN transmits voice in a circuit-switched manner with a transmission bandwidth of 64 Kb/s. A VoIP system uses digital signal processing technology to compress speech signals into 6.3Kb/s or lower, so the required bandwidth is greatly reduced, thus improving the efficiency of the physical link. In addition, the packet switching technology adopted by VoIP realises statistical multiplexing of channels and improves utilisation of network resources. The VoIP system has also improved the quality of voice calls, and its voice quality is not lower than that of Global System for Mobile communications (GSM) mobile phones, which meets the requirements of various users.

D. Conformity to the development trend of the integration of three networks

VoIP technology is in line with the trend of integration of telephone network, radio and television network and data network, and has a considerable market prospect.

2.2 Components of VoIP

The components of VoIP include: end-user equipment, network components, call

processors, gateways and protocols, as shown in Figure 2.1.

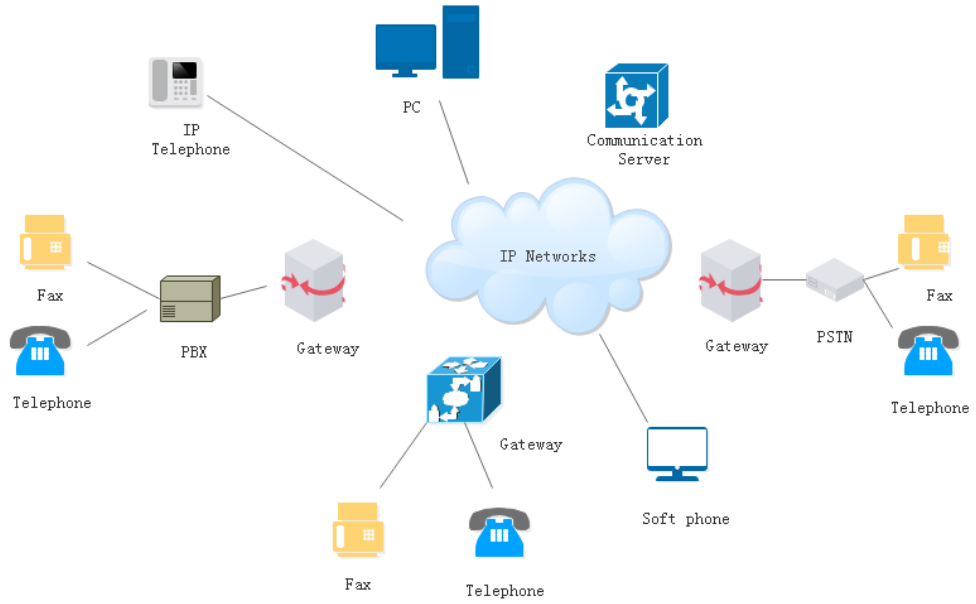

Figure 2. 1 Component of VoIP

2.2.1 End-user Equipment

End-user equipment is used for communications between a VoIP system and another terminal. The connection to the network on which VoIP runs can be either physical cable or wireless. The end user equipment can be a telephone on the desk or a softphone installed on a PC or mobile devices. Functions include voice and video communication, and may contain instant messaging, surveillance and monitoring functions (Ranch Network, 2004). Different types of user terminals generate voice data with various storage structures, so they need certain data conversion to form a unified IP packet to transmit on the same network. This process is usually accomplished by a gateway or an adapter in the VoIP system for data conversion. Developing terminals with uniform standards and specifications can reduce the cost

of data conversion.

2.2.2 Network Components

Network components include cables, routers, switches and firewalls (Ranch Network, 2004). Usually a new VoIP system is installed on an existing IP network，and its impact on the IP network is greater than merely adding more data traffic. The increased traffic is more urgent than most of the other supported data traffic to reach the destination. Switches, routers, and firewalls are needed to identify and act on VoIP data to ensure that the latency stays at a low level. Additional security measures such as encryption complicate the process.

2.2.3 Call Processor

Call processor functions include phone numbers to IP translation, call setup, call control, user authorisation, and signal coordination, which may help control bandwidth (NIST, 2004). Call processors are usually a piece of software running on popular operating systems. This leaves it open to possible network attacks, including the vulnerabilities of the given operating system, vulnerabilities of applications and other applications running on the operating system.

2.2.4 Gateways

The gateway mainly provides an interface between IP network and traditional PSTN

to realise VoIP voice communication. The gateway can support a variety of telephone lines, such as digital relay stack, analogue telephone line and PBX connection line. It can also offer voice coding compression, data structure conversion, signalling conversion, call control, dynamic routing calculation and other functions.

According to functionality, gateways are classified into three groups: signalling gateways (SG), media gateways (MG), and media controllers (NIST, 2004). The roles of gateways include handling call origination, detecting and conducting analogue to digital conversion. Signalling gateways are normally used to manage the signal traffic between an IP network that supports VoIP and a switching circuit network, while the media signals between these two networks are managed by media gateways. The media gateway controller is utilised to manage the traffic between signalling gateways and media gateways. MGCP (rfc2705) and Megaco are the most common gateway protocols that are used for VoIP. They are composites or derivations of previously ones but are now less used protocols (NIST, 2004).

The IP gateway is the core device of VoIP. The routing management function of the gateway maps the area code of each area to the gateway IP address of the corresponding area. When a user makes a call, the gateway maps the telephone area code to the corresponding IP address according to the database information, and adds this IP address to the IP packet, choosing the best route, so that the IP packet can reach the destination gateway through the network with a small delay. In areas where the Internet has not been extended or gateways have not yet been set up, voice communication over an IP network can be realised by setting up routing and connecting the nearest gateway through the telephone network.

2.2.5 Protocols

A VoIP system mainly uses three kinds of protocols: signalling protocol, media protocol, and IP protocol. The signalling protocol is used to establish connections between two communicating parties (terminals); the media protocol handles the real-time communication of audio or video; and the IP protocol is used for VoIP voice transmission.

2.3 VoIP Communications

2.3.1 Principles of VoIP Communication

VoIP transmits voice information on an IP network to realise real-time voice communication. As shown in Figure 2.2, the basic transmission process of VoIP is collecting the original sender's voice, converting the original voice signal into a digital signal by analogue-to-digital conversion, compressing and encoding the digital signal through a voice compression algorithm, encapsulating the compressed voice data according to the standard of TCP/IP, and sending the encapsulated IP packet to the receiver over an IP network (NIST, 2004). The receiver decodes and decompresses the received voice data packets to obtain the original analogue voice signal, so as to realise the real-time communication of voice information on the network.

The simplest form of VoIP system consists of two or more VoIP-enabled devices and IP networks, which are interconnected through IP networks. VoIP devices convert the analogue speech signals into IP data streams and send data streams to the

destination, which in turn convert IP data streams into analogue speech signals. The network between VoIP communication devices must support IP transmission, which can be any combination of IP routers and network links.

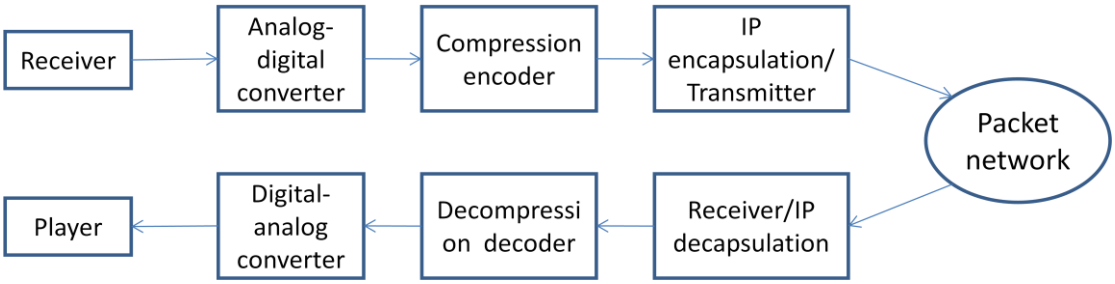

Figure 2. 2 The basic transmission process of VoIP

The transmission process of VoIP on the network is mainly divided into seven stages (NIST, 2004):

A. Analogue signal-digital signal conversion (ADC)

The original voice obtained by voice acquisition is an analogue signal, while the voice transmitted over an IP network should be digital signals. So the original voice signal must be converted to a digital signal through analogue-to- digital conversion, that is, digitalisation. Figure 2.3 shows the principle of PCM. Pulse code modulation (PCM) converts an analogue signal with continuous time and values into a digital signal with discrete time and values for transmission in the channel. Digital signals are generated by sampling, quantizing and coding the continuously changing analogue signals, that is, a pulse code modulation process. Since the pulse code modulation is particularly suitable for services requiring a large data transmission rate and high bandwidth, it is currently used in VoIP communication systems.

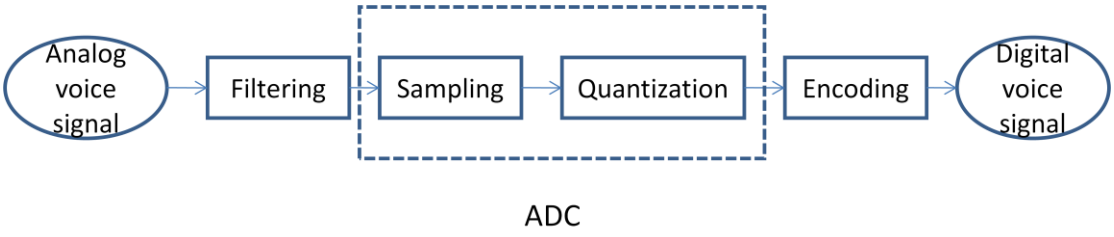

Figure 2. 3 The principle of PCM

B. Compression coding

Due to the limitation of the channel bandwidth, the voice signal needs to be compressed after being digitally encoded (NIST, 2004). Speech compression coding includes three forms, namely waveform coding, parameter coding and hybrid coding. Waveform coding is the direct conversion of waveform to digital code stream, which is divided into time domain waveform coding and frequency domain waveform coding. Parametric coding is to extract the characteristic parameters of speech from the transform domain of a speech signal, and then transform them into digital streams to compress and encode the speech signal. Hybrid coding is a combination of waveform coding and parameter coding. It has low bit rate in network transmission, good speech quality after decoding at the receiving end and moderate coding algorithm delay, but this coding method is more complicated. Most codecs have specific frame lengths, and voice packets are compressed according to the frame lengths. At present, there are commonly used speech coding standards such as G.729, IYU-T, and G.711. In a VoIP communication system, the voice coder of sender and receiver must use the same compression coding algorithm to ensure that the receiver can successfully recover the original voice signal.

C. IP data encapsulation

After the analogue-to-digital conversion and compression encoding, voice data packets enter the network processor, and need to be IP-encapsulated before they can be transmitted on the IP network (NIST, 2004). The network processor adds an IP header, timing and other information to the compressed voice packet, and then transmits the encapsulated IP packet to the receiving end over the network. An IP network typically adopts a connectionless User Datagram Protocol (UDP) transmission approach that does not form a connection. It requires data to be placed in variable-length datagrams or packets, accompanied by addressing and control information for transmission over the IP network. This method can reduce the time of establishing connection and transmission, and reduce the delay in the transmission process, thereby reducing the impact on the quality of VoIP voice communication.

D. Network transmission

In a VoIP communication system, voice data packets received from the sender are transmitted over an IP network to the receiver (NIST, 2004). The change of data packet transmission time on the network forms the jitter. The node on the network link checks the address of each voice packet before forwarding the datagram to the next station in the destination path route. The network links used by VoIP can be any access mode or topology that supports IP data streams.

E. IP data decapsulation

After receiving the IP datagram, the VoIP receiver’s device starts IP data decapsulation processing, removes the IP header and other information, and obtains the compressed encoded voice data packets.

F. Speech decoding

The original digitised voice data can be obtained by decompressing and decoding the voice packets after IP decomposition using the voice decoder with the same compression encoder algorithm as the sender.

G. Digital signal-to-analogue signal conversion (DAC)

The original analogue voice can be obtained by decoding the obtained digital voice data with the corresponding decoding algorithm of the sender, the voice data can then be transmitted to the sound card by the playback driver, and the original analogue signal can be obtained by the loudspeaker broadcast (NIST, 2004).

In short, voice data transmission on a VoIP network includes voice analogue signal to digital signal conversion, compression coding, IP encapsulation, network forwarding transmission, IP decapsulation and decompression decoding to restore the original analogue voice signal, so as to realise voice transmission on an IP network.

2.3.2 The Limitations of Best-Effort Service

The Internet Protocol provides a best-effort service which is to make its best effort to transmit datagrams from sender to receiver as quickly as possible. The Internet protocol is a connectionless transmission mechanism, which does not guarantee the quality of service. According to CISCO's Internetworking Technology Handbook, 'Quality of Service (QoS) refers to the capability of a network to provide better services to selected network traffic over various technologies, including Frames Relay, Asynchronous Transmission Mode (ATM), Ethernet and 802.1 networks, SONET, and

IP-routed networks that may use any or all of these underlying technologies (Cisco, 2004). The main factors that affect VoIP quality of service are packet loss, delay and jitter. In the process of VoIP transmission, the voice signal is transmitted to the receiver over an IP network after compression coding. Therefore, VoIP real-time voice communication applications are very sensitive to packet loss, delay and jitter. An IP network only provides the best-effort voice packet transmission, which cannot guarantee the quality of voice communications. Packet loss, delay and jitter have seriously impacts on the quality of voice communications using VoIP.

A. Packet Loss

A packet is lost either if it never arrives at the receiver or if it arrives after its scheduled playout time (Kurose & Ross, 2000). Packet loss is one of the important factors affecting the quality of VoIP communications. Packet loss caused by channel congestion, corrupted packets rejected and hardware failures may result in IP datagrams losing or arriving too late to reach the receiver. The retransmission mechanisms of TCP can reduce packet loss, but it is considered to be unacceptable in real-time voice communications such as VoIP applications due to the increase of delay. In addition, when packet loss occurs, the TCP congestion control mechanism will reduce the transmission rate of the sender and affect the voice intelligibility of VoIP. Therefore, a VoIP system usually runs over UDP and does not retransmit when packet loss occurs.

Packet loss is usually considered to be the interval in the communication process. In VoIP communication, according to the encoding and transmission mode of voice and the concealment technology of packet loss, a certain extent of packet loss can be

tolerated. Loss recovery schemes are used to preserve acceptable audio quality in the presence of packet loss. The common loss anticipation schemes include forward error correction (FEC), Interleaving and so on. Through forward error correction technology, some redundant information is transmitted along with the original voice information, so that some lost original voice information can be recovered from the redundant information, thereby reducing the impact of packet loss on voice quality. Interleaving can mitigate the effect of packet loss with no increasing bandwidth requirements of a stream (Kurose & Ross, 2000). In addition, changes in coding schemes and network tuning mechanisms can also help reduce packet loss (Mehta et al., 2004). When the network packet loss rate is too high, VoIP voice quality will be seriously affected, and the receiver will hear the voice with chaotic noise. In order to ensure the acceptable quality of VoIP communications, the random packet loss rate needs to be less than 5%.

B. Latency

Latency is defined as the time it takes for data to reach its destination from its source. The person who talks into the microphone is the data source, and the listener at the other end is the data destination. This is the one-way delay. Round trip delay is the total of the one-way delay and the time it backs to the originating client. In the US, PSTNs have a round trip delay of less than 150 ms (Mehta et al., 2004). According to the 1996 ITU, the delay limits of one-way end-to-end transmission of G.114 are:

Less than 150 ms: acceptable for most user applications and not perceived by a human listener;

150 to 400 ms: acceptable when the administrators are aware of the transmission time impact on the transmission quality of user applications;

More than 400 ms: unacceptable for general network usage (Mehta et al., 2004).

This means that the tolerable delay range for one-way delay is 75 ms to 400 ms. As the users of the US telephone system want to achieve a delay of less than 150 ms, 150 ms is assumed as the threshold of cumulative delay of VoIP system, and packets that are delayed by more than the threshold will be lost.

Different components of a VoIP system and the underlying IP network as well as other sources lead to delay. Constraints on the time of transmitting packets affect security solutions. Most security measures are combined by methods that are applicable to various parts of the VoIP system, which usually add delay. Encryption is one of the main concerns. Encryption and decryption have a significant impact on delay, which largely depends on the size of the key as well as the complexity of the algorithm. In general, the larger the key that is used to encrypt the data, the more secure the data is, but the more time it takes to encode and decode the data (Greenstreet et al., 2004). A balance between the desired security and the quality of VoIP communications must be struck. One of the greatest challenges of VoIP security solutions is to implement security solutions with minimal delay to ensure that quality of service meets organisational standards. Safety is often sacrificed for better quality.

C. Jitter

Due to the network status and other factors, the delay of data packets is a random queuing, which means the time from sender to receiver may fluctuate in different data

packets. This fluctuation in delay causes jitter.

Jitter can be caused by a long delay before the arrival of packets, which may lead to packet disorder. If the receiver in a VoIP system ignores the presence of jitter, it may cause the audio quality become unintelligible. As most VoIP communication is based on User Datagram Protocol, out-of-order packets cannot be re-assembled at the protocol level. However, with the support of the application, sequence numbers and timestamps enable the disordered packets to be re-assembled at the application layer to reduce jitter. It takes time to reassemble packets, and packets that need more time to arrive may cause packet loss to maintain the transmission of received packets.

The network layer can reduce the jitter generated by the network by creating a variable-length buffer, sorting the incoming data packets and transmitting them to the application program (Kuhn et al., 2004). The buffers can accommodate multiple voice packets, and users can choose buffers of different sizes according to their needs. In general, jitter decreases as the size of buffer increases, but when exceeding a certain size, it will lead to more packet loss. In the process of converting an IP packet into data, the decoder decompresses the encoded data packet and generates a new voice packet, which is sent to the decoding buffer after being operated in the same length as the decoder. 150 ms is used as the maximum latency for a VoIP system, the buffer needs to be cleared every 150 ms to reduce other latency. As some packets may delay longer than this, the buffer may be cleared before a group of packets is reassembled, leaving a gap, therefore resulting in packet loss. Another way to reduce jitter is to use the QoS features of routers, switches, and firewalls (Kuhn et al., 2004).

The best-effort service of an IP network imposes limitations of packet loss, delay, and

jitter on VoIP communications. In VoIP communication, the delay must be less than 400 milliseconds, the jitter should not exceed 50 milliseconds, and the packet loss rate should not exceed 5% to ensure the quality of service of VoIP (Rizal, 2014).

2.3.3 Key Technologies of VoIP

Since the best-effort connectionless service provided by the traditional IP network cannot provide the guarantee of service quality, there are limitations such as packet loss, delay and jitter. VoIP real-time voice service has higher requirements on these parameters, some key technologies must be adopted to ensure the quality of service of VoIP.

The key technologies of VoIP include coding technology, real-time transmission technology, quality of service (QOS) guarantee technology, network transmission technology, signalling technology, echo cancellation technology and so on.

A. Coding technology

Voice compression is an essential technique for interactive voice communication systems such as VoIP. Voice compression reduces the network bit rate or bandwidth on the communication channel (Nagirireeddi, 2008). Since IP networks have limitations on network bandwidth, compression rate is one of the important parameters of voice in VoIP systems. VoIP uses a variety of compression coding techniques to minimise the bandwidth requirements of communications. The main coding technologies currently used include G.711, G.722, G.729 and G.723 defined by ITU-T. G. 729 and G. 723 are used to reduce the bandwidth requirements of VoIP

systems. G.729 can compress the sample speech from 64Kb/s to 8Kb/s with virtually undistorted quality (Nagirireeddi, 2008). In packet switching networks, the quality of service cannot be guaranteed. The speech coding needs adaptability of coding rate and coding scale, which makes the speech coding more flexible. Therefore, the standard of G.729 speech coding has been extended from 8 Kb/s to 6.4 Kb/s ~ 11.8 Kb/s, and the quality of speech has changed with the change of speech coding. At the lowest bandwidth of 6.4Kb/s, the voice quality is acceptable, so G. 729 is suitable for a VoIP communication system. G.723 is the lowest rate speech coding algorithm that has been currently standardised (Nagirireeddi, 2008). It uses 5.3/6.3 Kb/s dual-rate speech coding, which has higher speech quality but larger delay. G.722 coding is used to improve speech quality and provide better speech perception than PSTN. In addition, mute detection is also one of the key technologies for reducing bandwidth in VoIP. Mute detection can effectively eliminate the silent signal, so that the occupied bandwidth of speech signal can be further reduced to about 3.5 Kb/s (Nagirireeddi, 2008). In codec selection, the main parameters to be considered are bit rate, quality, delays, and complexity (process and memory) requirements (Nagirireeddi, 2008).

B. Real-time transmission technology

As real-time voice communication, VoIP requires real-time transmission technology. The real-time transmission technology mainly uses the Real-time Transport Protocol (RTP). RTP is a protocol that provides end-to-end real-time data transmission including audio (Nagirireeddi, 2008). RTP includes two parts: media data and control information such as time parameters. The control part is called RTCP. RTCP is used to convey the end-to-end quality of the data stream in an RTP session (Nagirireeddi,

2008). Statistics including packet loss rate, delay, jitter and number of packets sent help the session monitor the connection status. Voice in a VoIP system is sensitive to delay, and RTP can help the end-to-end real-time voice to be delivered properly. RTP can provide time tag and synchronisation mechanism to control data flow. It enables the receiver to reconstruct the sender's data packet and provide quality of service feedback for the receiver. It provides an important real-time transmission technology for VoIP.

C. Quality of Service Assurance Technology

VoIP mainly uses Resource ReSerVation Protocol (RSVP) and Real-time Transport Control Protocol (RTCP) to control network congestion and provide quality of service guarantee. The Resource ReSerVation Protocol is a protocol for quality integration services on the Internet. RSVP allows hosts to request special quality of service over the network for the transmission of special application data streams. The real-time transmission control protocol is used to monitor the quality of service and ensure the quality of VoIP calls.

D. Network transmission technology

The network transmission technologies used in VoIP mainly include Transmission Control Protocol (TCP), User Datagram Protocol (UDP), gateway interconnection technology, network management technology, routing selection technology, charging technology and security authentication technology. VoIP uses RTP to provide end-to-end real-time data transmission services. The RTP header contains the identifier, a sequence number, transmission monitoring and timestamp of the payload

data. According to the reasons mentioned in Section 3.3.2 of this chapter, VoIP applications usually use TCP to establish reliable connections in the call establishment phase, while UDP is used to transmit voice fast in the voice communication phase. In VoIP systems, UDP packets are usually used to carry RTP protocol data units. Shorter data units can reduce latency. IP, UDP and RTP headers are calculated according to the minimum length, and the VoIP voice packet overhead is very large. The VoIP format using the RTP protocol inserts multiple voices into the voice data segment in this manner, thus increasing the transmission rate (Nagirireeddi, 2008).

E. Signalling Technology

Signalling technology ensures the smooth implementation of telephone calls and voice quality (Nagirireeddi, 2008). The signalling protocols used by VoIP mainly include H.323 series of ITU-T and the Session Initiation Protocol (SIP) of the IETF.

H.323 is a protocol suite set up by the International Telecommunication Union (ITU)，which provides the foundation for IP-based real-time communications such as audio, video and data communications (IEC, 2004). It specifies the components, protocols, and procedures providing multimedia communications over packet-based networks. Various configurations of video, audio and data exist in H.323. There are some possible configurations, such as audio only, audio and video, video and data, as well as audio, video and data. H.323 does not specify packet networks and transport protocols (IEC, 2004).

H.323 defines four types of network components: Terminals, Gateways, Gatekeepers,

and Multi-point Control Units (MCUs) (Kamara & Kohno, 2006). Terminals refer to the end-user equipment. Gateways use protocol conversion and media format conversion to process communications between different networks. Gatekeepers provide a number of services: addressing, authorisation and authentication, as well as accounting and call routing. MCUs handle conferencing. The ITU specifies an H.323 zone consisting of terminals, gateways, Multi-point Control Units, and a gatekeeper that manages this zone.

H.323 protocols include H.320 for ISDN, H.321 for B-ISDN and H.324 for PSTN terminal (Nagirireeddi, 2008). H.323 provides interoperability between devices, between high-level applications, and between providers. It does not rely on network structure, independent of operating system and hardware platform, and supports multipoint functional multicast and bandwidth management.

The process of establishing H.323 call includes three kinds of signalling, namely Registration Admission Status (RAS) signalling, H.225 call signalling and H.245 control signalling. RAS signalling is used to process registration, authorisation, bandwidth change, status and disconnection between terminals and gatekeepers (Nagirireeddi, 2008). H.225 call control signalling is used to establish a connection between two H.323 terminals. H.245 signalling channel is built between two terminals or a terminal and a gatekeeper. It is mainly used to transmit control messages between them, including capacity exchange, mode parameter request and control information.

Due to the reliability and easy management of H.323 protocol set, it is widely used in multimedia communication on the network. However, H.323 also has some limitations,

such as not supporting multi-cast multicast conferences, and does not support call forwarding. It takes a long time to establish a call.

Session Initiation Protocol (SIP) is a signalling protocol that was developed by the Internet Engineering Task Force (IETF), which is used to set up and close two-way communications sessions (Goode & Bur, 2002). SIP operates on the application layer and it is used with a number of different protocols. TCP can provide higher security with SSL/TLS, whereas UDP can achieve faster and lower latency connections.

The main components of an SIP system include a user agent (UA), a proxy server, a registrar server and a redirect server. The user agent software includes the client and the server components. The client is responsible for outgoing calls, and the server ensures that incoming calls are received. The proxy server forwards data traffic, the registrar server is responsible for authenticating requests, and the redirect server is used to resolve information for the user agent client (Qiu & Qi, 2003).

The endpoint starts by connecting with a proxy and / or a redirect server which is responsible for resolving the destination number to an IP address. Then it sends back the information to the originating endpoint that transmits information directly to the destination. One of the security advantages of SIP is that it uses only one port (Kuhn et al., 2004).

For SIP, the main security concerns are information integrity, confidentiality, authentication, non-repudiation, and privacy. No new security measures have been specially designed for SIP, and SIP uses the security technologies offered by the Hypertext Transfer Protocol (HTTP) for web pages, the Simple Mail Transfer Protocol

(SMTP) for email systems, and Internet Protocol Security (IPSec) for the network layer.

Full encryption provides the best signal confidentiality, but as some proxies have to read and/or modify some SIP message fields, other possible methods should be considered. If the proxy is trustable, the best security solution is to perform encryption at the transport layer and/or the network layer. IPSec technology is used to realise full packet encryption at the transport layer and the network layer. TSL had been used but has been deprecated later on (Ramsdell, 2004). Full encryption requires that every end point can support the encryption approach.

The 401 and 407 response codes as well as header fields are used in HTTP authentication. It provides a stateless challenge-based scheme for authentication, in which the challenge and user credentials are transmitted in the header fields. When a proxy or a user agent receives a request, it may issue a challenge to confirm the sender's identity. Once the identity is confirmed, the recipient also needs to verify that the requester has been authorised (Ramsdell, 2004).

For email systems, the early Pretty Good Protocol (PGP) has been replaced by Multipurpose Internet Mail Extensions (MIME), which is then extended to Secure/Multipurpose Internet Mail Extensions (S/MIME). S/MIME can enhance security for SIP because MIME bodies are carried by SIP. MIME contains components that are able to provide integrity and encryption for MiME data (Kuhn et al., 2004). As described in rfc2633, S/MIME can be used for 'authentication, message integrity and non-repudiation (using digital signatures) and privacy and data security (using encryption)' (Ramsdell, 2004). As network components need to use data in the

header field, S/MIME is very useful if full encryption of the package seems impossible.

User identification is accomplished by comparing the certificate belonging to the user with the header information. The verification of the integrity of the message is to match the information in the external header field with that in the internal header field. In general, the Session Description Protocol (SDP) is encrypted using S/MIME, but it may require the encryption of certain header components. The header privacy in SIP is achieved by means of MIME type messages/SIP to encapsulate the entire message. For anonymous purposes, it needs to decrypt messages before the certificate to be identified and subsequently validated (Ramsdell, 2004).

F. Echo Cancellation Technology

Echo is one of the important factors that affect the quality of VoIP voice communication. In a telephonic voice conversation, echo is the return of a person's speech with delay, with a reduced or modified sound level, and with a certain amount of distortions (Nagirireeddi, 2008). Echoes in voice communication include acoustic echo and electrical echo. Acoustic echo is generated through an acoustic medium of a speakerphone or hands-free phone functionality. Electrical echoes are normally created in the two-to-four-wire telephone conversation hardware hybrids (Nagirireeddi, 2008). Echo cancellation technology mainly uses digital filter technology to eliminate the echo interference, which has a great impact on the call quality and guarantee the call quality (Nagirireeddi, 2008). In IP packet based networks with large delay, echo cancellation technology is particularly important.

VoIP uses real-time voice transmission, which easily results in poor voice quality. Among the many factors that affect voice quality, echo is one of the most critical factors. Echoes in a VoIP system include the echo of the speaker and the echo of the listener. The speaker's echo refers to hearing her or his own voice during the conversation, and the receiver's echo refers to hearing the speaker's voice repeatedly. When the echo coincides with the original voice, it does not affect the normal conversation. However, when the echo does not coincide with the original voice, the listener can easily perceive the existence of the echo, thus affecting the quality of voice calls. Packet switching technology is used in voice transmission over the Internet. The encoding, compression and packaging of voice signals cause large echo delay and jitter. In order to improve the quality of voice, echo processing is required in the process of voice transmission on the Internet. The IP Telephony Gateway must have an echo cancellation function. Devices such as echo suppressors can be used to reduce the impact of echo on the quality of the call.

In a VoIP system, voice samples from the sender are compressed using compression codecs such as G.711, G.722, G.729 and G.723, and then framed as payloads. The size of the payload varies with the compression codec, compression rate option, and payload duration (Nagirireeddi, 2008). The compressed payloads are framed as RTP/UDP/IP packets and sent on the IP network. In real VoIP systems, voice payload, RTP, RTCP, quality of service (QoS) mechanisms, voice quality monitoring, bandwidth management and other parameters work together to provide better end-to-end packet delivery services over the Internet.

2.4 Security Analysis of VoIP Communications

Since VoIP reduces the cost of audio communication and promotes the utilisation of network resources, it has a great application value in long-distance communication and enterprise internal communication, thus it has flourished in the field of telecommunications. However, most VoIP communication systems are operated on public networks, especially on the Internet, they are therefore vulnerable to various security threats on the network they run (Yildiz et al., 2016; Chikha et al., 2016; Vera-del-Campo et al., 2015). Common VoIP security threats include DoS attacks, eavesdropping and tampering.

This section will discuss the security issues and possible countermeasures of VoIP according to the VoIP security taxonomy defined by Voice over IP Security Alliance (VoIPSA).

In 2005, the Voice over Internet Protocol Security Alliance (VOIPSA) developed a classification of VoIP security (Zar et al., 2005). This Taxonomy defines many potential security threats to VoIP deployments, services, and end users (Zar et al., 2005). VOIPSA classifies VoIP security threats into six categories: social threats, eavesdropping, Interception and Modification, Service abuse, Intentional Interruption of Service, and Other Interruptions of Service.

The main security issues of VoIP include Network Address Translation, denial of service, database, Web server, additional VoIP services provided by vendors, protocol stack, access to public and unknown networks, physics and power and so

on.

NAT allows one network address to be translated into another address at a gateway between two networks so that the packet has a valid source address on the network it is on (Tucker, 2007). In general, NAT (rfc1621) is used to translate a private IP address into a public Internet routable IP address. Ports may also be converted. NAT is normally only a concern if end-user devices connect directly with an external network or if they connect to an internal network from an external network (Kuhn et al., 2004).

NAT is regarded as a layer of security because it conceals a real address on the internal network from the public network (Kuhn et al., 2004). As the routing device used may not know the actual IP address of the end-user device, NAT can become a problem. The header field contains the information that defines the endpoint. So routing devices must be capable of reading the header field, and in some cases (i.e. when using proxy firewall) altering it (Kuhn et al., 2004). However, this is hampered when using encryption.

The desirable solution is not to use NAT if possible. Without using NAT, the problem mentioned above is solved, although other problems may arise. When NAT is needed, it must take care when selecting applications and proxy firewalls that realise the implementation, and public network services are alternatives.

Denial of service is caused by preventing the service from being delivered (Tucker, 2007). Denial of service can be caused by unavailable bandwidth or unavailable VoIP components. There are many factors that can lead to a DoS; for example, the

network is too congested to provide the bandwidth required to support the application; the server cannot handle the traffic; the operation of unrelated services reduces available service resources; malicious programs, e.g. viruses and Trojans; other malicious programs aiming at causing DoS; or hacking activities (NIST, 2004).

For a denial of service as a result of bandwidth constraints, a possible solution is to increase bandwidth and / or isolate VoIP traffic to get priority service (Tucker, 2007). There are various ways that can be used to prevent the server from not working, helping reduce the denial of service caused by component failures, such as clustering. The components of the VoIP system supplied by the vendor should be evaluated, so as to remove any unnecessary parts. The size of the server should be planned in advance to support the desired services and expected traffic, therefore increasing the percentage of expected growth.

It is difficult to defend against malicious programs and malicious activities, but appropriate patches should be applied in a timely manner and virus protection should be installed with frequent updates. In addition, installation designers should consider host-based firewalls, intrusion detection software or intrusion prevention suite (Tucker, 2007).

An excellent approach to defending against DoS attacks of public servers is to locate the device with publicly available IP addresses behind a firewall or other device that only allows communications from trusted sources (Tucker, 2007). In addition, measures such as strengthening the operating system in use, removing all unnecessary services and applications from servers and workstations, and patching can also effectively defends against the DoS.

Other concerns of a VoIP system that need to focus on include Web servers, databases, protocol stacks, additional VoIP services provided by vendors, access to public and unknown networks, physical security, electrical power, and so on.

When storing and retrieving the required information, some nodes of the VoIP component need database to complete various functions of the VoIP system (Rosenberg et al., 2004). It should apply the database security principle, such as changing the default administrator password, patching when there are patches available, and accessing to the database, particularly from sources rather than from the VoIP system.

A web browser is a common feature of end-user devices, and it used to offer additional functionality and increased productivity (Tucker, 2007). A VoIP server can have a web browser interface that provides management functionalities (Kuhn et al., 2004). It needs to patch the device when a new patch is available and use as strong authentication as possible.

As each vendor has its own implementation of VoIP, it may require any number of devices to run on a server to support its product (Tucker, 2007). Patches should be updated in real time and all unnecessary services should be turned off. If there is a great risk, it should apply encryption and / or consider protection measures by other equipment, such as a firewall. As voice applications and operating systems have similar vulnerabilities, they should also be patched frequently.

When a VoIP system locates within a secure network and only has access to the public network through a network gateway, the gateway is a vulnerability that needs

to be considered. So a hardened gateway should be deployed behind the appropriate firewall, and the gateway is aware of the protocols used.

VoIP must deal with the protocol it supports, so it needs some network protocol stack implementation. Detailed instructions about how to implement protocol stacks are normally provided by vendors or purchased from other vendors. For the latter, the same vulnerabilities are inherited from a specific vendor's protocol stack purchased (Collier & Mark, 2004). When patches are available, patching is necessary.

The components of a VoIP system should be physically secure. The ownership specifies permission to gaining access to the component. With physical access, there are many ways to compromise a VoIP device depending on the device used and the underlying operating system. Good security measures include removing a disk and CD-ROM option from the boot list and using a password to protect the configuration.

Denial of service occurs when a component is not available. A separate power source and an uninterruptible power supply (UPS) should be in place to defeat the DoS in case of power loss.

In a word, VoIP system security should start with solid security on the internal network (Tucker, 2007). It should be protected against threats from attached hostile networks and those from internal networks as well. The security policy should consider any specific VoIP requirements. The payload of the VoIP system should be adapted to the relevant network and server to ensure that the appropriate resources are available. Carrying out a risk analysis of each component and process can identify vulnerabilities and threats. This provides the information that is required to determine

appropriate measures. Making a balance between security and the business needs of a organisation is key to the success of the VoIP deployment (Tucker, 2007).

2.5 Summary

This chapter explores how VoIP communications work and discusses the security issues that may occur in the VoIP communication process. VoIP technology has many advantages such as low cost, extensibility, better use of network resources and so on, thus it has become one of the most popular communication technologies in the world.

The components of VoIP include end-user equipment, network components, call processors, gateways and protocols. Each component of VoIP works in a coordinating way to provide high quality of service.

The basic VoIP transmission process is that a VoIP device converts the analogue speech signals into IP data streams and sends the data streams to the destination, which in turn converts the data streams into analogue speech signals. Since the Internet Protocol network provides a best-effort service which does not guarantee the quality of service, VoIP communication faces the problems such as packet loss, delay and jitter. Key technologies such as coding technology, real-time transmission technology work together to provide better end-to-end packet delivery service.

Security issues could occur in each component of VoIP. The Voice over Internet Protocol Security Alliance developed a classification of VoIP security. The main security issues of VoIP include network address translation, denial of service, database, Web server and so on, and some countermeasures could help defend VoIP against malicious attacks to which VoIP communications might be subjected.

According to the principles and security analysis of VoIP communication, the effect of packet loss, latency and jitter needs to be addressed in the process of covert VoIP communications, as well as security issues such as passive adversary and man-in-the-middle attacks to provide the availability and security of VoIP communication.

CHAPTER 3 Steganography

The security concerns of computer and network systems have traditionally been addressed using tools from cryptography. Cryptography is a mature field with decades of development and steganography is the little and much younger sister of cryptography. Digital steganography is regarded as a promising technology to complement cryptography in addressing data protection issues. This chapter presents the principles of digital steganography. It starts with a description of computer and network security, followed by defining steganography and identifying its main components. The classification of steganographic methods and steganographic evaluation criteria are also briefly described in this chapter. The attacks on steganographic communication are examined to show the importance of the current research. The chapter provides a summary and synthesis of related work in streaming media steganography.

3.1 Computer and Network Security

Computer security is the protection afforded to an automated information system in order to attain the applicable objectives of preserving the confidentiality, integrity, and availability of information system resources, including hardware, software, firmware, information/data, and telecommunications (Guttman & Roback, 1995). These three

concepts are often referred to as the CIA triad (Figure 3.1). The three concepts embody the fundamental security objectives for both data and computing services.

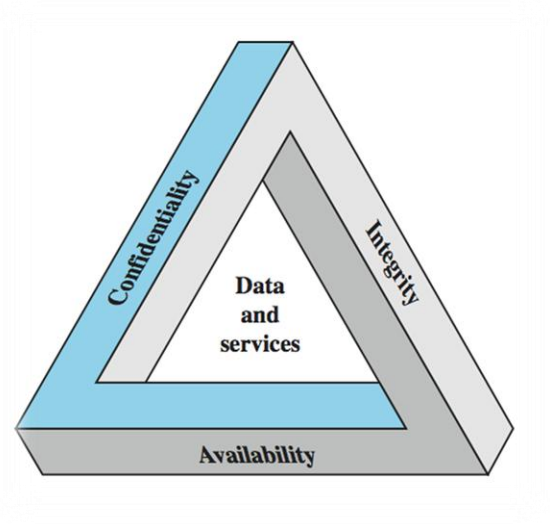

Figure 3. 1 CIA triad (Guttman & Roback, 1995)

Confidentiality. Confidentiality is one of the most important properties of secure data and service, which preserves authorised restrictions on information access and disclosure, including means for protecting personal privacy and proprietary information. A loss of confidentiality is the unauthorised disclosure of information (FIPS 199, 2004).

Confidentiality covers both data confidentiality and privacy (Stalling & Brown, 2018). The former ensures that private or confidential information is not made available or disclosed to unauthorised individuals. The latter ensures that individuals can control or influence what information related to them may be collected and stored and by whom that information may be disclosed.

Confidentiality is the protection of transmitted data from passive attacks. With respect to the content of data transmission, several levels of protection can be identified. The broadest service protects all user data transmitted between two users over a period of

time. For example, when a TCP connection is set up between two systems, this broad protection prevents the release of any user data transmitted over the TCP connection. Narrower forms of the service can also be defined, including the protection of a single message or even specific fields in a message (Stallings & Brown, 2018). These refinements are less useful than the broad approach and may even be more complex and expensive to implement. The other aspect of confidentiality is the protection of network traffic flow from analysis. This requires that an attacker be unable to observe the source and destination, frequency, length, or other characteristics of the traffic on a communications facility.

For network security, confidentiality means only the sender and intended receiver should be able to understand the contents of the transmitted message (Kurose & Ross, 2000). Because eavesdroppers may intercept the message, this necessarily requires that the message be somehow encrypted so that an intercepted message cannot be understood by an interceptor. This aspect of confidentiality is probably the most commonly perceived meaning of the term secure communication. Cryptographic techniques for encrypting and decrypting data can be used to provide data confidentiality.

Integrity. The integrity of data and system includes guarding against improper information modification or destruction, and ensuring information non-repudiation and authenticity. A loss of integrity is the unauthorised modification or destruction of information (Stalling & Brown, 2018).

Integrity covers two related concepts, namely data integrity and system integrity (Stalling & Brown, 2018). The former assures that information (both stored and in

transmitted packets) and programs are changed only in a specified and authorised manner, and the latter assures that a system performs its intended function in an unimpaired manner, free from deliberate or inadvertent unauthorised manipulation of the system.

As with confidentiality, integrity can apply to a stream of messages, a single message, or selected fields in a message. The most useful and straight forward approach is total stream protection (Stallings, 2017).

A connection-oriented integrity service, one that deals with a stream of messages, assures that messages are received as sent with no duplication, insertion, modification, reordering, or replays (Stallings, 2017). The destruction of data is also covered under this service. Therefore, both message stream modification and denial of service are addressed in the connection-oriented integrity service. On the other hand, a connection less integrity service, one that deals with individual messages without regard to any larger context, generally provides protection against message modification only.

There is a distinction between service with and without recovery. The integrity service relates to active attacks and is concerned with detection rather than prevention (Stallings, 2017). If a violation of integrity is detected, then the service may simply report this violation, and some other portion of software or human intervention is required to recover from the violation (Stallings, 2017). Alternatively, there are mechanisms available to recover from the loss of integrity of data. The incorporation of automated recovery mechanisms is, in general, the more attractive alternative.

The communicating parties want to ensure that the content of their communication is not altered, either maliciously or by accident, in transit. Extensions to the check summing techniques that are encountered in reliable transport and data link protocols can be used to provide such message integrity (Kurose & Ross, 2000). Digital signatures and end-point authentication are important cryptographic tools of providing message integrity.

Availability. The availability of data and services ensures timely and reliable access to and use of information. A loss of availability is the disruption of access to or use of information or an information system (Stalling & Brown, 2018). Availability assures that systems work promptly and service is not denied to authorised users.

Both X.800 and RFC 4949 define availability to be the property of a system or a system resource being accessible and usable upon demand by an authorised system entity, according to performance specifications for the system (i.e. a system is available if it provides services according to the system design whenever users request them) (Stallings, 2017). A variety of attacks can result in the loss of or reduction in availability. Some of these attacks are amenable to automated countermeasures, such as authentication and encryption, whereas others require some sort of physical action to prevent or recover from loss of availability of elements of a distributed system.

An availability service is one that protects a system to ensure its availability. This service addresses the security concerns raised by denial-of-service attacks. It depends on proper management and control of system resources and thus depends on access control service and other security services.

3.2 Introduction to Steganography

Information hiding is the use of human sensory insensitivity and the redundancy of the multimedia digital signal itself, to hide information in public host signals, such as image, audio, video or text document, without affecting the sensory effects and use value of the host signal (Cox et al., 2008). Currently, the branches of information hiding technology include covert channel, steganography, anonymity and copyright marking, and the two main branches of which are steganography and digital watermarking. Steganography is the act of concealed communication. The very existence of a steganographic message is secret (Cox et al., 2008).

Electronic communication is increasingly subject to cyber attacks such as eavesdropping, tampering and malicious interventions. The information transmission security of electronic communication systems is mainly relied on cryptography, which includes encryption technology and data integrity authentication technology (Shen et al., 2007). Currently, encryption technology is facing some new challenges. On the one hand, with the enhancement of network computing capabilities, some encryption algorithms have been cracked. For example, the MD5 algorithm was successfully cracked by Wang et al. (Wang et al,. 2005).. On the other hand, as the information content is encrypted, its cryptographic form is a kind of exposure of the existence of the confidential information, thus the encryption method is easy to become the main target of attack.

In the early 2010s there was an explosion of interest in steganographic systems for

the embedding of various digital contents in the field of information transmission security. Steganography is a method of embedding the secret data into a cover object, which should not cause unacceptable distortion and arouse observers' suspicions (Tang et al., 2014). Both steganography and encryption technology protect the confidentiality of the secret data, but there are significant differences in many aspects between them. The fundamental difference between steganography and the traditional encryption technology is that the encryption technology hides the "content" of the secret data, while steganography hides the "existence" of the secret data. The encryption technology emphasises that the secret data is changed to 'beyond understandable', whilst steganography emphasises that the secret data is changed to 'invisible', such that unauthorised users know neither the existence of the secret data nor the details of it. Compared with the encryption technology, steganography does not attract the attacker's attention and reduce the probability of being attacked. Therefore, steganography is considered to be one of the most important technologies for information transmission security.

In recent years, steganography has become a flourishing area of research with more and more practical applications in various fields. For example, military communication systems usually need to be protected at higher security levels. They require not only encrypting the messages exchanged, but also hiding the existence of the secret communication, which means attackers even cannot perceive the existence of the exchanged messages. To protect the intellectual property of digital products, merchants often embed their trademark or unique logo into digital products with steganography. There are also some other applications of steganography.

In early steganography literature, steganography was widely used in image (Yang et al., 2008; Lee & Chen, 2000; Marvel et al., 1999), audio (Darsana & Vijayan, 2011; Cvejic & Seppanen, 2002), and video files (Cetin et al., 2012). For image steganography, the common method was to modify the least significant bit (LSB) of pixels in an image using LSB-based algorithms. Since the Human Visual System (HVS) is not so sensitive, the differences between the original cover image and the stego image with a hidden secret message are imperceptible by human eyes. Audio steganography was used to embed a secret message into an AU, WAV, or MP3 audio file. It is generally recognised that audio steganography is more challenging than image steganography for the wider dynamic range of the Human Auditory System (HAS) in comparison with HVS. The basic LSB steganography was also widely used in audio and video files. With the rapid spread of the Internet applications, steganography has been increasingly applied to streaming media on the Internet.

Many western universities and research institutes have vigorously conducted steganography research, such as Harvard, MIT, and Stanford University in the United States, UCL, University of Cambridge, and the University of Surrey in the UK, Otto-von-Guericke-Universität in Germany, the University of Toronto in Canada, and so on, as well as many large companies such as IBM, NEC, etc. The initial steganographic methods focused on the most common type of data hiding in storage-based multimedia files. Many research institutions and universities in the developing world are engaged in this field, such as the Key Laboratory of Information Security of Chinese Academy of Sciences, Beijing Institute of Electronic Technology and Applications, Shanghai Jiao Tong University, University of Science and

Technology of China, Beijing University of Posts and Telecommunications, Xidian University, Harbin Institute of Technology, Dalian University of Technology, Sun Yat-sen University, and so on.

Research in network channel based steganography has yielded some landmark results. For example, Ahsan and Kundur first reported their studies of steganography in the TCP/IP header in 2002 (Ahsan & Kundur, 2002); Murdoch et al. carried out information hiding in ISN field in the TCP header on Linux and Open BSD (Murdoch & Lewis, 2005); Li et al. studied steganography based on the MAC network protocol (Li & Ephremides, 2005); Later, substantial effort has been directed to the most commonly used TCP/IP and other network protocols respectively (Zander, Armitage & Branch, 2007; Sellke et al., 2009), resulting in a variety of steganographic algorithms that embed the secret data in the protocol headers. Researchers at the Beijing University of Posts and Telecommunications and the Hunan University also conducted in-depth research in steganography using network channels; Researchers from the CAS Software Institute put forward a steganographic algorithm that embed the secret data in the TTL field in the IP header (Qu, Su & Feng, 2004).

At present, although encryption technology is facing severe challenges, it is still the main means of securing information transmission on the network; in contrast, steganography technology has a slight disadvantage in terms of theoretical research, technology maturity, and practicality, but its potentiality to address security issues remains strong, especially in solving the key issues in the secure transmission of massive data on the network. It is believed that it will play an important role in the future information and communication system.

Steganography technology, also known as covert communications technology, has not yet been developed into a mature and practical stage. There are still key technical issues that need to be addressed, such as the enhancement of data embedding capacity, the fusion of steganographic algorithms, the data security storage and retrieval of encrypted data. Effective methods for hiding secret messages in static image and audio files (Cheng & Huang, 2001; Yuewei Dai et al., 2001) prompted further research in steganography for streaming media.

3.3 Components of Steganography

The first informal definition of a steganographic scheme was formulated by Simmons as the Prisoners' Problem (Simmons, 1983). Two prisoners, Alice and Bob, are the surveillance of a warden, Eve. The warden permits Alice and Bob to communicate, but all communications must go through the warden. If the warden thinks that Alice's message to Bob is innocuous, she may simply forward it to Bob. Alternatively, she may intentionally distort the content (e.g. apply lossy compression) in the hope that such a distortion will remove any secret message that might be present. If the warden thinks Alice's message to Bob hides a covert communication, then she may block the communication entirely.

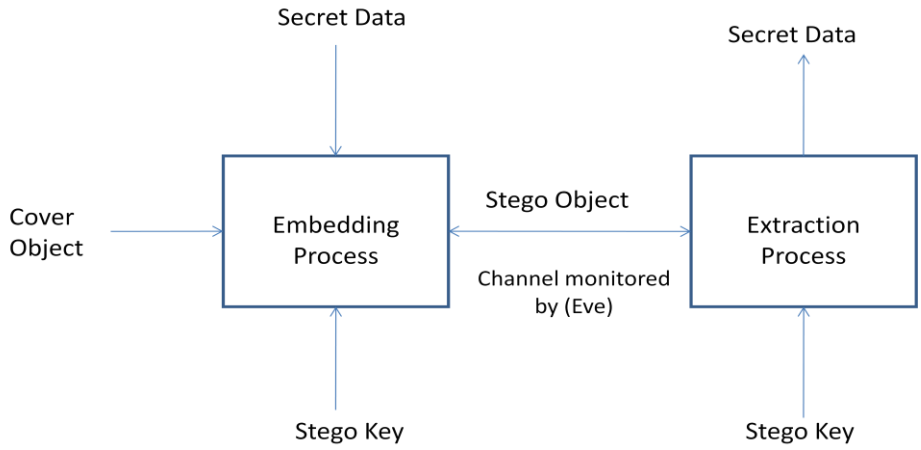

Figure 3. 2 Steganographically embedding scheme

This framework is depicted in Figure 3.2. The components of steganography include cover object, secret data, stego key, embedding process, stego object and extraction process. A number of different assumptions can be made regarding the channel, the source of cover object, and the embedding and extraction processes.

3.3.1 Cover Object

The cover object is an object in which the secret data is embedded (Cox et al., 2008). Unlike a digital watermark, a steganographic message is not related to the cover object in which it is hidden. Consequently, the steganographer is free to choose a particular cover object from the source of covers. The cover object can be an image, video clip, audio clip or other digital media, and the secret data is embedded in the redundancy in their representation format. The cover object has a specific embedding capacity depending on the media type and the embedding method, and the likelihood of being detected varies with different cover objects. For example, it is intuitively clear that noisy or highly textured images will better mask any embedding changes than high-quality images with little content (e.g. blue sky images).

In fact, it is possible to go a step further and choose the cover object such that it is correlated with the secret data. For example, if the secret data is an image, choose an image cover that is similar. The advantage of this is that the minimum number of bits needed to encode the secret data is now the conditional entropy of the secret data given the cover. This may be very much smaller than the entropy of the secret data itself. As a result, the fewer bits that are need to hide the secret data, the less likely the warden will detect the stego object. Furthermore, the information transferred by Alice to Bob can now be much greater than the number of bits embedded. This is because the cover object is now providing Bob with additional information.

3.3.2 Secret Data

The secret data could be any stream of binary representation that needs to be transmitted over an insecure channel without raising suspicion (Cox et al., 2008). The amount of secret data that can be embedded in the cover object depends on the capacity of the steganographic system. Generally, the probability of detection increases with the amount of the embedded data.

3.3.3 Embedding Process

The embedding process usually has three inputs: cover object, secret data, and an optional stego key. It uses a particular method, for example LSB replacement, to embed the secret data into the cover object and create the stego object as an output.

Fundamentally, an embedding function can be based on three different principles,

namely (Cox et al., 2008):

A. The cover objects are preexisting and the embedder does *not* modify the cover objects. This is referred to as steganography by cover lookup.

B. The cover objects are generated based on the hidden message and the embedder does *not* modify the cover objects. This is referred to as cover synthesis.

C. The cover objects are preexisting and the embedder modifies the cover objects. This is referred to as steganography by cover modification.

The type of changes introduced by the embedder, together with the location of these changes within the cover object, have a major influence on how inconspicuous the embedded message is (Cox et al., 2008). Intuitively, changes of large magnitude (great percentage change) will be more obvious than changes of small magnitude (Fridrich, 2014). Consequently, most steganographic schemes ought to modify the cover object as little as possible.

The location of the changes is controlled by the *selection rule*. There are three types of selection rules: sequential, (pseudo) random and adaptive (Cox et al., 2008).

A sequential selection rule embeds the message bits in individual elements of the cover object in a sequential manner (Cox et al., 2008), for example, starting in the upper left corner of an image and proceeding in a row-wise manner to the lower right corner. Although the sequential selection rule is the easiest one to implement, it provides poor security, since steganalytic algorithms can inspect the statistical properties of pixels in the same order, looking for a sudden change in behaviour.

A pseudo-random selection rule embeds the message bits in a pseudo-randomly selected subset of the cover object (Cox et al., 2008). The sender might first use a secret stego key to initialise a pseudo-random number generator (PRNG) that in turn generates a pseudo-random walk through the cover object. The message bits are then embedded into the elements constituting this walk. Pseudo-random selection rules typically offer better security than sequential rules.

An adaptive selection rule embeds the message bits at locations that are determined based on the content of the cover object (Cox et al., 2008). The motivation for this is that statistical detectability is likely to depend on the content of the cover object as well. For example, it will be more difficult to detect embedding changes in noisy images or in highly textured areas of the image compared with smooth, uniform areas. Thus, one may desire to adjust the selection rule to the specific content of the cover object. For example, consider LSB embedding once more. The selection rule could depend on the variance of pixels within a small local neighbourhood. Only pixels whose local neighbourhood variance exceeds a certain threshold would be candidates to be modified. Thus, the selection rule has influence on the security of steganographic schemes.

3.3.4 Stego Object

The stego object is the object containing a steganographic message, i.e., the modified version of the cover object after embedding the secret data (Fridrich, 2014). The stego object should be similar to the original cover object and maintain the

property of imperceptibility.

3.3.5 Stego Key

The stego key is a secret key used in the embedding process to make the secret data computationally infeasible to be extracted by the extraction process without having access to that secret key. The key can be used for several different purposes. As previously mentioned, the key may seed a pseudo-random number generator to generate a random walk through the cover object (Chandramouli, Kharrazi, & Memon, 2004). It can also be used to generate other pseudo-random entities needed for embedding or just a password for decoding the embedding location.

The covert message may be cryptographically encrypted prior to embedding, in which case, there is also a crypto key (Cox et al., 2008). While the stego and crypto keys could be derived from one master key, it is simpler to consider the two keys separately.

It is important to choose strong stego keys; otherwise the warden could attack the steganographic scheme simply by trying to read messages from the stego object using all possible stego keys (Cox et al., 2008). The correct key could be revealed when a meaningful message is obtained. Although this attack would not work if the message was encrypted before embedding, there exist more advanced versions of this attack (Fridrich et al., 2005). In general, it is always a good practice to encrypt the message before embedding. The crypto key could be derived from the stego key or could be chosen independently. The second choice provides better security in cases

where the stego key is compromised.

3.3.6 Extraction Process

The extraction process is an opposite process of the embedding process mentioned above. It takes the stego object and an optional stego key as an input and extracts the secret message as an output. According to the requirement of the original cover object, it can be divided into blind extraction and informed extraction.

3.4 Classifications of Steganographic Methods

There are some methods of classifying steganography. Since each type of steganography has its own properties and attributes, it is very useful to classify steganographic systems. The main classification approaches are based on cover type, embedding domain and extraction/steganalytic condition (Cole & Krutz, 2003; Peng et al., 2020; Cox et al., 2008).

3.4.1 Based on the Cover Type

As many different types of digital media can be used as cover objects for embedding secret data, steganographic methods can be classified based on the type of the cover object (Cole & Krutz, 2003).

According to cover types, steganography can be divided into three categories: the first category is static steganography in image, audio, text and other digital files; the

second one is network steganography based on modification of the network protocols; and the third is streaming media steganography which embeds secret data into streaming media (such as VoIP audio stream, MPEG2 video stream, etc.).

In early steganography literature, steganography was widely used in static covert objects, such as text, image, audio, and video files. Some other effort has been directed to network steganography. So far, the first two types of steganography research is in their mature stages, and the covert steganographic communication that uses streaming media as cover object is becoming a hot area of research. Some security experts anticipate that, advances in streaming media steganography will set off a climax of information security technology in the next few years.

Streaming media steganography has attracted the attention of information security experts all over the world. On the one hand, streaming media contain plenty of redundancy, which can be used to embed secret data without causing signal distortion. Compared with image, audio, text and other digital files and network channels, streaming media are regarded as a better candidate for the cover object. On the other hand, due to the broad application of streaming media on the network, such as digital television on broadcast television networks, IP television (IPTV) on the Internet, Video on demand (VoD), Audio on demand (AoD), Audio and video mail, Video telephony, and Voice over IP (VoIP), streaming media steganography will have extensive application prospects in the field of information transmission security.

3.4.2 Based on Embedding Domain

According to the embedding domain, steganography can be divided into time/space domain steganography, transform domain steganography and compressed domain steganography (Cox et al., 2008).

In terms of image and audio cover objects, time/space domain steganography is implemented by modifying the grey values or intensity values of image pixels or audio signals. The algorithm is simple, but it is difficult to resist common image or audio processing and attacks, and its robustness is poor.

The transform domain algorithm refers to embedding secret data into the transform domain coefficients of the original carrier, mainly including Discrete Fourier Transform (DFT) domain, Discrete Cosine Transform (DCT) domain, Discrete Wavelet Transform (DWT) domain, and so on (Peng et al., 2019). The transform domain algorithm has obvious advantages. Firstly, the conventional processing of the cover object can be regarded as low-pass filtering, and the algorithm usually avoids adding secret data to the high-frequency part to resist the influence of compression and low-pass filtering. Secondly, the embedding of secret data is distributed to the whole carrier to resist geometric attacks more effectively. Thirdly, the parameter distribution in the transform domain usually makes the embedding of secret data more in line with human perception characteristics.

Compressed domain steganography is usually closely associated with multimedia standards. The multimedia used for cover object in steganography usually has the international or industry standards. In order to be compatible with the formats set by

these standards, it is necessary to embed and extract the secret data in these formats (normally compressed formats). Common compression domains include JPEG compression domain, MPEG compression domain, MPEG-2 compression domain, JPEG2000 compression domain and so on.

3.4.3 Based on Extraction/Steganalysis Condition

Blind extraction and blind steganalysis are divided into the following four indicators: extraction does not require the original cover object; extraction does not require the original secret data; steganalysis does not require the original cover object; steganalysis does not require the original secret data. Blind steganography should satisfy the following conditions: neither the original cover object nor the original secret data is needed for steganalysis or extraction; otherwise it is called non-blind steganography (Cox et al., 2008). Therefore, according to whether the original cover object is needed during extraction and steganalysis, steganography can be divided into blind extraction, informed extraction, blind steganalysis and targeted steganalysis.

The concept of blind or informed message extraction is absent from the steganography literature. It is usually implicitly assumed that the recipient of the covert message does not have the original cover object available for use in decoding. In practice, however, this need not be the case (Cox et al., 2008). For example, both Alice and Bob may have the same database of images that they agree to use as cover object. In this case, the use of informed extraction algorithms may be beneficial

as the embedder does not need to embed the covert message as strongly as possible. Consequently, the risk of an adversary detecting the covert communication will be smaller.

All steganalysis algorithms can be categorised as either system attacks, targeted attacks, or blind attacks (Cox et al., 2008). System attacks use a fault in the implementation or weaknesses due to an insufficient stego key space. A steganalysis algorithm is described as blind if the method of detection is independent of the steganographic method used. Conversely, a targeted steganalysis method is one that is designed to detect the stego object that is created by one or more particular steganographic methods (e.g. LSB embedding).

In blind methods, the object is usually represented in some high-dimensional feature space. Machine learning methods are then used to distinguish between the clusters of cover object and stego object in the feature space. An important advantage of blind methods is that they can potentially detect an unknown stego scheme. They are also capable of classifying the stego object to individual steganographic methods.

Because targeted steganalytic methods need to be designed for each individual steganographic method, their construction cannot be automated. On the other hand, such methods may be more accurate than blind methods.

3.4.4 Others

In addition to the main classification criteria mentioned above, there are other classification criteria. For example, according to the hiding method, steganography

can be divided into insertion based, substitution based, generation based, and cover lookup based steganography; according to whether the embedding is reversible, it can be divided into reversible steganography and irreversible steganography; according to whether the selected secret data is single or multiple, it can be divided into single steganography and multiple steganography; according to the anti-attack capability, it can be divided into robust steganography, vulnerability steganography and semi-vulnerability steganography.

3.5 Properties and Evaluation Criteria of Steganography

The primary goal of steganography is to hide the fact that a covert communication is present within an innocuous communication (Cox et al., 2008). There are some properties and criteria that can be used to evaluate steganographic systems.

3.5.1 Undetectability

Since the main propose of steganography is to hide the very existence of the secret data, it is significant to consider the property of undetectability of steganographic systems. Undetectability is the impossibility of detecting the presence of steganographically embedded data in a cover object (Cox et al., 2008). Many factors can directly affect the undetectability, such as the choice of the cover object and the embedding algorithm.

Perfect undetectability means that an adversary with unlimited computational power is not able to state if in a given overt communication there is also a hidden message

transmitted. Although perfect undetectability may not exist in practical applications, it is of great value to attain better undetectability.

3.5.2 Imperceptibility

Imperceptibility refers to the fact that embedding secret data in a digital cover object should not degrade the perceived quality of the cover object and should ensure that the normal use of the stego object is not affected (Cox et al., 2008). The common measurement methods are subjective evaluation method and objective evaluation method. Subjective evaluation methods rely on people's subjective judgment on the quality of the cover object before and after embedding the hidden message. Many steganographic methods use the limitation of Human Visual System (HVS) or Human Auditory System (HAS) in the data embedding process (VenkatramanS, Ajith, & Paprzycki, 2004). At present, objective evaluation methods are mostly based on errors, that is, to examine the signal-to-noise ratio of the stego object and the cover object.

3.5.3 Security

Security is another important property of steganography. There are three types of warden for steganography: passive, active and malicious (Cox et al., 2008). Currently, most steganographic methods are designed for the passive warden scenario in which the warden just passively observes the communication and does not interfere with the communication in any way. An active warden may introduce distortion with the goal to

prevent steganographic communication. For example, a stego object might be compressed or down sampled to fit in with a given bandwidth. The malicious warden may intentionally try to remove the hidden message or impersonate the communicating parties.

Security means that the embedded data should be confidential. In many applications, information hiding requires security. Adversaries cannot know that there is secret data embedded in the digital cover object, let alone remove the secret data. The common method is to scramble or encrypt the embedded data or its location. It requires using one or more keys in the processes of embedding and extracting the secret data. For example, in early data hiding algorithms, pseudo-random signals are embedded as hidden objects. In this case, the seed value of pseudo-random signal generator is the key. Secure steganographic algorithms have the following requirements: It is impossible to remove the secret data. This requires keeping the embedding location distribution of the secret data secure; Non-authorised user cannot extract the secret data. This requires that the secret data itself be encrypted or scrambled.

3.5.4 Capacity

There are two kinds of capacity in relation to the field of steganography: embedding capacity and steganographic capacity. The embedding capacity is the maximum number of bits that can be embedded in a cover object using a given steganographic system (Cox et al., 2008). The steganographic capacity is the maximum number of bits that can be hidden in a given cover object, such that the probability of detection

by an adversary is negligible (Cox et al., 2008). Therefore, the steganographic capacity is likely to be much less than the embedding capacity.

Most steganographic schemes can avoid being detected by current steganalysis simply by decreasing the amount of information embedded in a cover object. Alternatively, decreasing the number of embedding changes to the same payload will, in general, decrease the embedding impact and thus lead to a more secure scheme. However, practical steganographic schemes must have usable steganographic capacity. While a robust, 1-bit watermark may be very useful for applications, a steganographic scheme that can communicate only 1 bit in an image is not practical. According to the International Federation for the Phonographic Industry (IFPI), the steganographic capacity of 20 bits/second is required for audio steganography. Thus, the primary goal of new steganographic algorithms is to develop statistically undetectable methods with high steganographic capacity.

The steganographic capacity is susceptible to the following factors: Different models of cover object produce different capacity expressions. A number of capacity analysis focuses on extracting the features of image cover object and abstracting the probability distribution of the cover object; attack is the main factor causing capacity loss. Almost all capacity analysis results take into account the impact of attacks; whether the algorithm is blind extraction can also impact the steganographic capacity. Usually, blind extraction requires a certain loss of steganographic capacity.

3.5.5 Robustness

Robustness refers to the fact that the hidden information can remain intact and can be extracted accurately when it is distorted by various conventional signal processing or malicious attacks (Cole & Krutz, 2003). The robustness of a steganographic method is the resistance to changes of a hidden message during transmission. The robustness is often defined as modifications caused by typical operations on the carrier during transmission. However, it could also include deliberate modifications introduced by a potential adversary in order to hinder steganographic communication. Generally speaking, different steganographic applications require different robustness.

Two factors can affect the robustness of steganographic systems. The first one is undetectability, and the second is the ability to defeat the active attack (Wang & Wu, 2004). The secret data should be recovered by the second party if the cover media faced some data processing. A steganographic method could be considered as robust if both the detection and the destruction of the hidden data are hard.

The absolute robustness of all possible attacks and combinations of these attacks are difficult to obtain because of the variety of attack methods. However, a successful robust steganographic system should ensure that the value of the protected data has also been destroyed when the hidden secret data is compromised.

The main goal of improving steganographic systems is to enhance the requirements of undetectability, imperceptibility, security, capacity and robustness. However, sometimes enhancing a certain property may negatively affect others. For example,

high undetectability may normally mean low capacity. Therefore, there should be a trade-off among these properties according to practical applications.

3.6 Attacks on Steganographic Communication

The primary goal of steganalysis is to detect when a covert communication is occurring (Cox et al., 2008). The requirements for steganalysis can vary significantly depending on the operational scenarios. As mentioned above, there are three types of attack on steganographic communication: passive, active, and malicious.

A passive attacker intercepts communications from the sender and tests for the presence of a hidden message (Cox et al., 2008). If no secret data is detected, then the attacker forwards the communications to the receiver. Conversely, if a hidden message is detected, then the attacker blocks the transmission to the receiver (i.e. the receiver receives neither the communication nor the hidden message).

An active attacker is permitted more freedom. In particular, the active attacker is free to modify the communication between the sender and the receiver. Thus, even if the steganalysis test fails, the communication may still be altered in an attempt to remove any hidden message that might evade detection (Chandramouli et al., 2004). For example, if Alice and Bob are transmitting JPEG compressed images between one another, then the warden, Eve, might choose to recompress the images before forwarding them to Bob. In this way, any hidden message embedded with a JPEG-based steganographic algorithm is likely to be removed.

A malicious attacker may go one step further, in that the attacker might attempt to

impersonate the sender and send the receiver false messages (Cox et al., 2008). To do so requires the attacker to be capable of much more than the detection of a covert message. The attacker must also know what steganographic algorithm the sender and receiver are using and any associated steganographic and encryption keys (Chandramouli et al., 2004).

Whilst the fundamental goal of an attacker is to reliably detect the presence of a hidden message within an innocuous communication between the communication parties, the attacker may want to know much more. This higher level of analysis, which is more than just detection, is known as forensic steganalysis. Forensic steganalysis aims at identifying the steganographic method and the stego key and recovering message attributes such as the message length or content (Cox et al., 2008).

There are two major classes of detection algorithms: targeted and blind. Targeted steganalysis is intended to detect a specific (target) steganographic algorithm, while blind steganalysis aims to detect a wide range of steganographic algorithms, including previously unknown algorithms. Both targeted and blind steganalysis algorithms are classification problems and consequently results from pattern recognition and machine learning can be applied.

Detection of steganalysis algorithms can also be accomplished at a system level by exploiting weaknesses in the implementation of specific steganographic algorithms. For example, sequential LSB embedding can be reliably detected using the histogram attack. Sample pairs analysis is an advanced steganalytic method targeted at LSB embedding along a pseudo-random path. Both methods can estimate the length of

the embedded message. Calibration is a method for constructing detection statistics for both targeted and blind steganalysis. It is based on estimating the cover object from the stego object and is appropriate when the embedding is performed in the compressed (DCT) domain (Cox et al., 2008). For detection of steganographic algorithms that embed data in the spatial domain, the histogram characteristic function (HCF) can be a useful feature (Cox et al., 2008). In the spatial domain, noise reduction filters form the equivalent of calibration. The performance of steganalysis algorithms can vary considerably depending on the source of the cover objects.

3.7 Steganography in Streaming Media

Many research efforts are devoted to covert communication over streaming media such as VoIP. Existing covert VoIP communications stem from two research origins and are divided into two main types. The first one comes from digital media steganography, which modifies the digital representation of the transmitted voice to embed secret data in different processing stages. This type of covert VoIP communications is called voice payload-based steganography. The second one is VoIP-specific protocols-based steganography, which embeds secret data into specific VoIP protocol fields or their parameters.

3.7.1 Voice Payload Approach

Least significant bit (LSB) steganography is one of the most popular methods for covert communications. Meanwhile, there are also some techniques that can be

applied to other VoIP steganographic methods to enhance their undetectability, robustness, and / or steganographic bandwidth.

A. LSB methods applied to digital voice signals

The first VoIP steganographic method that used the digital voice signal as a hidden data carrier was proposed by Aoki in 2003 (Aoki, 2003). The LSB algorithm provided a Packet Loss Concealment (PLC) method for G.711-based VoIP. Later on, the PLC method was further improved, but the steganographic method and voice codec remained the same (Aoki, 2004; Aoki, 2007).

The prototype of the first VoIP steganography implementation that was put forward by Dittmann et al. in 2005 also used the LSB method (Dittmann et al., 2005). This work was further extended in 2006 by Krätzer who proved that a typical VoIP communication could be practically used for steganography (Krätzer et al., 2006).

Many researchers attempted to improve LSB steganography. In 2006, Wu and Yang described an adaptive LSB scheme (Wu et al., 2006). The results showed that this method outperformed simple LSB and provided a higher steganographic bandwidth (about 20 Kbps) with less degradation of voice quality. The SteganRTP implemented by Druid in 2007 used the least significant bits of the G.711 codec to carry steganograms, and provided a reliable bi-directional covert communication channel which allowed to exchange 1 kB/s of secret data in single direction (Druid, 2007). In 2007, Wang and Wu also proposed using the least significant bits of voice samples to carry secret data, but the steganographic bits were encoded using a low-rate voice encoder, i.e. Speex (Wang et al., 2007). This implementation was characterized by a

small processing delay of about 0.257 ms.

A proof-of-concept implementation of LSB-based Voice over VoIP (Vo^2IP) was presented by Takahashi and Lee in 2007 (Takahashi et al., 2007), which established a hidden communication by embedding 8kbit/s G.729-based compressed audio data into the conventional Pulse Coded Modulation (PCM)-based audio data streams. Later on, Liu et al. found that the least significant bit of each speech frame of G.729 could be replaced with a secret data bit (Liu et al., 2008). The method was perceptually transparent when the steganographic bandwidth was relatively high (about 200 bit/s).

There has been an explosive growth in research into LSB based covert VoIP communication. In order to recover the secret data loss due to network conditions, Huang et al. described how to implement covert VoIP communication using LSB matching (Huang et al., 2008). They developed a G.711-based prototype called Stega-Talk. It was based on the Redundant Audio Data RTP payloads that are typically used to load Dual-Tone Multi-Frequency (DTMF) digits.

In 2008, Tian et al. proposed an application based on a LSB steganographic system, which employed a balanced and simple confidential information encryption (Tian et al., 2008). The system was evaluated for VoIP with G.729a speech coding using a proof-of-concept tool named StegTalk. The achievable steganographic bandwidth was in the range 0.8-2.6 kbit/s with a negligible effect on speech quality. In addition, it met the real-time requirements of VoIP services.

In order to eliminate the correlation of secret data and enhance its ability to resist

statistical analysis, Tian et al. brought forward a real-time VoIP steganographic system in 2009 (Tian et al., 2009a). The scheme utilized M-sequence encryption technology to resist statistical analysis. Moreover, the protocol steganography (using free / unused protocol headers) was used to provide a new synchronisation mechanism together with the RSA-based key protocol to ensure accurate restitution of the secret data on the receiver side. The system produced Mean Opinion Score (MOS, usually used to express VoIP call quality) 0.3 and 1 quality declined under the 0.8 and 2.6 kbt/s steganographic bandwidth. When transmitting 1 MB of steganogram, the latency increased by approximately 4.7 ms. Similar to the LSB-based method, Tian et al. presented the adaptive VoIP steganographic method in the same year (Tian et al., 2009b).

In 2009, Xu and Yang proposed an LSB-based speech transmission method using G.723.1 speech codec in 5.3kbt/s mode (Xu et al., 2009). They defined the five least significant bits of the Line Spectrum Pair Vector Quantization (LSP VQ) indices used to transmit secret data; this method provided a steganographic bandwidth of 133.3 bit/s.

Moreover, some methods have taken adaptation into account. Tian et al. described a Dynamic Matrix Encoding Strategy (DMES) to dynamically select the size of each message group in a given set of adaptable message sizes (Tian et al., 2010). The purpose of DMES was to flexibly adapt the steganographic bandwidth and embedded transparency according to user requirements. The main advantage of DMES is probably that it coded and covered independently.

Miao and Huang introduced an adaptive steganographic scheme based on the

smoothness of speech blocks (Miao et al., 2011). The security of the algorithm was improved by selecting a lower embedding rate in a relatively smooth voice blocks and selecting a higher embedding rate in sharp ones. In terms of speech quality, this algorithm is better than the classical steganographic algorithms based on LSB. About 7.5kbit/s of secret data was sent covertly with a degradation of voice quality of less than 0.5 on MOS scale.

For the adaptive balance between steganographic transparency and steganographic bandwidth, Tian et al. presented an Adaptive Partial-Matching Steganography (APMS) approach (Tian et al., 2011a). They introduced a Partial Similarity Value (PSV) measure to evaluate the partial matching between secret information and public information. In addition, three message sequences were used to eliminate the correlation between secret information, guide the adaptive embedding process, and encrypt synchronous signals. Later on, an insightful overview of the general techniques that can be applied to VoIP steganography to make it more difficult to detect was conducted by Tian et al. (Tian et al., 2011b). In addition, they proposed three new encoding strategies based on digital logic. All techniques were evaluated for LSB-based steganography and proved to be effective.

In 2011, Xu et al. brought forward another LSB-based adaptive steganography algorithm named Adaptive VoIP Steganography (AVIS) (Xu et al., 2011). AVIS had two components: Value-based Multiple Insertion (VAMI) and Voice Activity Detection Dynamic Insertion (VADDI). VAMI was used to dynamically select multi-bits based on VoIP vector values, and VADDI dynamically changed the embedding interval to make detection harder. The method was implemented for G.711-based VoIP, achieving

lower detectability than a classical LSB method when the steganographic bandwidth was about 114 B/s, and the generated latency was acceptable degrading the voice from 0.1 to 0.4 on MOS scale.

Instead of the LSBs, Liu et al. adopted the least-significant-digits to hide secret data (Liu, 2012). This method could increase the steganographic bandwidth by about 30% and bring lower steganographic costs than the classic LSB algorithm.

B. Non-LSB steganographic methods

In addition to the LSB-based steganography approaches, there are some other effective methods to use audio payload as a secret information carrier. These solutions are usually based on:

a. Phase encoding (Takahashi et al., 2007; Nutzinger et al., 2011),

b. Quantization Index Modulation (QIM) technique (Xiao et al., 2008),

c. Spectrum techniques in the transform domain (Takahashi et al., 2007;Nutzinger, 2010),

d. Echo hiding technique (Takahashi et al., 2007),

e. Analysis by synthesis (ABS)-based scheme (Ma et al., 2007; Wu et al., 2009),

f. Discrete Fourier Transform (DFT) technique (Deng et al., 2008),

g. Speech codec-specific approaches (Aoki, 2008; Aoki, 2009; Aoki, 2010; Geiser et al., 2008; Nishimura, 2009; Huang et al., 2011b).

For phase encoding-based hiding approaches, Nutzinger and Wurzer (Nutzinger et al., 2011) introduced a new method of speech phase encoding in 2011. Usually, the original phase value was replaced by some random data. In this scheme, the original

phase values were preserved to ensure higher speech quality. The algorithm embedded confidential data by introducing a phase difference of the blocks in a configurable phase spectrum. It achieved up to 12.5bit/s while introducing almost no degradation on the voice signal, good robustness and security.

Moreover, Quantization Index Modulation (QIM) technique is an effective method. In 2008, Xiao et al. (Xiao, 2008) introduced an interesting algorithm for QIM, which could be applied to low bit-rate speech streams. Li et al. (Li et al., 2012) proposed Complementary Neighbour Vertices-Quantization Index Modulation (CNV-QIM). The algorithm was based on dividing the codebook into two parts, representing “0” and “1” respectively. Moreover, the Complementary Neighbour Vertices (CNV) algorithm was used to determine the relationship between codewords. This guaranteed that each codeword was the opposite of its nearest neighbour, thus giving a bound of distortion. Experiments for Internet Low Bit Rate Codec (iLBC) and G.723.1 speech codec proved that this method was effective as it only slightly reduced the voice quality and provided a steganographic bandwidth of 100 bits/s.

There are also some steganography approaches based on spectrum techniques in the transform domain and echo hiding technique. Apart from the LSB algorithm, Takahashi and Lee (Takahashi et al., 2007) also took into account the feasibility of other algorithms that could be used for VoIP steganography, such as Direct Sequence Spread Spectrum (DSSS), Frequency-Hopping Spread Spectrum (FHSS), or echo hiding. All three algorithms maintained good voice quality and robustness when the hidden bandwidth was about 20bit/s. In 2010, Nutzinger et al. further improved DSSS, which created a hybrid steganography algorithm, combined DSSS with frequency

hopping and bit rate (Nutzinger, 2010). The authors also implemented a prototype and pointed out that the impact of the proposed scheme on voice quality should not be neglected.

As for analysis by synthesis (ABS)-based scheme, Ma et al. (MA et al., 2007) adopted an ABS algorithm to hide the MELP voice information of 2.4 kbit/s in G.721-based voice carrier in 2007. The obtained steganographic bandwidth was estimated up to 8 kbit/s. In 2009, Wu et al. (Wu et al., 2003) also used ABS to construct a steganography scheme based on Linear Predictive Coefficients (LPCs) as the secret data carried by means of LPCs substitution. For four speech coding methods (G.721, GSM, G.728 and G.729), the proposed method provided high steganographic bandwidth (800-3600 bit/s) and offered good undetectability, robustness and real-time performance. The method was superior in comparison with four traditional information hiding techniques, namely, LSB, echo hiding, phase coding and spectrum transform.

Discrete Fourier Transform (DFT) technique can also be used in information hiding. In 2008, Deng et al. (Deng et al., 2008) proposed the concept of Covert Speech Telephony (CST) to provide secure covert voice communications. The scheme used a robust Discrete Fourier Transform (DFT) watermarking scheme to hide speech in G.711-based VoIP data streams. Its main merit lied in using speech recognition to effectively reduce the size of secret information that was encrypted and embedded into existing overt VoIP calls.

Many researchers have engaged in speech codec-specific approaches. In 2008, Aoki (Aoki, 2008) presented a steganographic algorithm based on PCMU (PCM μ-law) feature, in which the 0-th speech samples could be represented by two codes due to

the overlap (namely +0 and -0). This redundancy was used to embed secret information into speech without causing voice degradation. Depending on the noise level of the background, this method achieved a steganographic bandwidth of 4.4-24 kbits/s. In 2009, Aoki (Aoki, 2009) published an extension of the work, studying the lossless steganographic technology of G.711PCMU and DVI-ADPCM codecs. The scheme also achieved redundancy in codec folding binary encoding, embedding hidden information without degrading speech quality. According to the background noise level, the improved technology provided the bandwidth from 24 to 400 bit/s based on G.711 calls, and the ADPCM calls were upgraded from 0 to 8bit/s. In 2010, Aoki (Aoki, 2010) proposed a semi-lossless variant technique to increase the steganographic bandwidth.

In 2008, Geiser et al. (Geiser et al., 2008) reported that the PLC algorithm was mainly used in wireless VoIP systems, which relied on the information of the specific side of the covert communication channel. In order to achieve secret transmission, the Algebraic Code-Excited Linear Prediction (ACELP) codebook or the fixed codebook (FCB) was divided into subcodebooks to uniquely identify the selected secret information bits. The prototype implementation based on Adaptive Multi-Rate (AMR) achieved a 2kbit/s steganographic bandwidth.

In 2009, Nishimura (Nishimura, 2009) described an interesting study in which information was hidden in the AMR-coded stream by using an extended quantisation-based method of pitch delay (one of the AMR codec parameter). This additional data transmission channel was used to transform the audio bandwidth from a narrow band (0.3-3.4 kHz) to a broadband (0.3-7.5 kHz).

In 2011, Huang et al. (Huang et al., 2011b) proposed a high-capacity steganographic scheme based on the utilisation of inactivity frames of G.723.2 speech codec. The authors proved that inactive frames in VoIP data streams were more suitable for embedding data than active frames, thus more data could be embedded in them with the same imperceptibility. They then proposed using a steganographic algorithm in different speech parameters of the inactivity frame for G.723.1 codec with 6.3 kbit/s bit rate. The scheme was imperceptible and reached an embedding rate of 101 bits/frame. Later in 2012, Huang et al. (Huang et al., 2012) published another research paper “Steganography Integration into a Low-bit Rate Speech Codec” in IEEE Transactions on Information Forensics and Security, which proposed a covert communication method to integrate steganography functions into speech compression coding. In the low bit rate speech coding process, the information was embedded in the pitch estimation, and the information hiding and speech compression were synchronised to complete.

In March 2014, Tang et al. (Tang et al., 2014) published a research paper entitled “Audio Steganography with AES for Real-time Covert Voice over Internet Protocol Communications” in the international academic journal *Science China Information Sciences*, which was recommended by the Chinese Computer Federation (CCF). The theory and method of real-time VoIP covert communication based on AES was proposed, and the corresponding VoIP steganographic algorithm based on variable embedding capacity and AES encryption was used to realize real-time covert VoIP communication.

In 2016, Tang et al. (Tang et al., 2016) published a research paper entitled “Universal

Steganography Model for Low Bit-Rate Speech Codec" in a CCF recommended international journal "Security and Communication Networks", which presented a universal model for low bit voice coded covert communication. The PSEQ degradation value and decoding errors were used as the basis to select the suitable steganography algorithm for each low bit speech coding, which made it easier for other researchers to realize secure covert communication with low bit rate speech coding.

3.7.2 VoIP Specific Protocol Approach

There are three main types of steganographic methods related to VoIP-specific protocols, methods modifying protocol data unit (PDU)'s time relations, methods modifying PDU – protocol specific fields, and hybrid methods.

A. Methods modifying PDU's time relations

The methods that modify PDU's time relations take the delay between frames in particular packets into consideration. In 2005, Wang et al. (Wang et al., 2005) first presented using the VoIP protocol as a steganographic carrier; in 2006, Chen et al. (Chen et al., 2006) also described. The authors proposed to embed a 24-bit digital watermark into the encrypted data stream (such as Skype, etc.) to track its propagation over the network, thus providing de-anonymisation. The digital watermark was inserted by modifying the delay between frames in particular packets in the VoIP data stream. By choosing the parameters of digital watermarking, they achieved a 99% true positive and 0% false positive rate while maintaining good

robustness and undetectability. The steganographic bandwidth of around 0.3 bit/s was sufficient for the description application; however, it was relatively low for covert communication.

In 2006, Shah et al. (Shah et al., 2006) inspected the use of jitter injected into VoIP packets to create a covert channel. This method intended to obtain the user's keyboard activity, etc., and it was confirmed that the attack was feasible even if the VoIP data stream was encrypted.

In order to find a suitable method for covert VoIP communication, Shah and Blaze (Shah et al., 2009) suggested a new information hiding technology called Interference Channel, which caused external interference to shared communication media (such as wireless networks) to send secret information. The wireless interference channel they described in the implementation of the 802.11 network transmitted secret information in the data stream with a lower steganographic bandwidth (about 1 bit per 2.5 seconds of the call), and was proved to be suitable for the VoIP data stream.

B. Methods modifying PDU – protocol specific fields

Instead of modifying PDU's time relations, some researchers focus on the PDU's protocol fields. In 2006, Mazurczyk and Kotulski (Mazurczyk et al., 2006a) suggested the use of steganography in the unused fields of the RTP protocol header, using digital watermarks to embed additional information into the RTP data stream to provide information source authentication and content integrity. The necessary information was embedded in the unused fields of the IP, UDP, and RTP protocol headers as well as the transmitted audio. The authors further improved their

approach (Mazurczyk et al., 2006b), combining Real-Time Transport Control Protocol (RTCP) functionality without the need to separate protocol functions, saving the bandwidth used by VoIP connection.

To seek a broader network steganographic method applicable to VoIP, Mazurczyk and Szczypiorski (Mazurczyk et al., 2008a; Mazurczyk et al., 2008b) proposed a method that hided information into the signalling protocol, SIP with SDP, and RTP streams (also with RTCP). They argued that the combination of steganographic schemes could provide about 2,000 bits of secret information transmission capacity during the signal connection phase and 2.5 kbits/s during the conversation phase. In 2010, Lloyd (Lloyd, 2010) further extended the algorithm, introduced steganographic algorithms for SIP and SDP protocols, and verified the feasibility through real experiments.

Moreover, some approaches are based on the header field of protocol. In 2008, Bai (Bai et al., 2008) proposed a covert channel based on the jitter field of the RTCP header. The method consisted of two stages: first, calculated the jitter value in the current network jitter domain. Then, the secret information was modulated into the jitter domain according to the parameters calculated before. Using this modulation method ensured that the statistical characteristics of covert channels were similar to those of the public channels. In 2009, Forbes (Forbes, 2009) introduced an RTP-based steganographic algorithm that modified the value of the timestamp in the RTP header field to send a secret message. The theoretical maximum steganographic bandwidth of the method was estimated to be 350 bit/s.

In 2010, Wieser and Roning (Wieser & Roning, 2010) implemented VoIP

steganography on Session Border Controller (SBC), and used it as the gatekeeper of trust boundaries. It was to find out whether SBC had some countermeasures for information hiding technology based on SIP and RTP protocols. A steganographic bandwidth of up to 569 kB/s was achieved.

Other researchers used the Network Time Protocol (NTP) time field in RTCP's Sender Report (SR). In 2011, Huang et al. (Huang et al., 2011a) described how to provide efficient key distribution in a VoIP steganographic communication environment. The proposed algorithm was based on the use of the NTP field in RTCP as a steganographic carrier, achieving 54 bit/s steganographic bandwidth while having good undetectability.

The payload domain of RTP packets can also be used to hide information. In 2011, Mazurczyk et al. (Mazurczyk et al., 2011) suggested the transcoding steganographic (TranSteg) method that relied on the compression of overt data in the payload domain of RTP packets to provide free space for hidden information. In the transcoding algorithm for the specified audio stream, an encoding method achieved similar speech quality in the case of a less audio load than usual. The voice audio stream was then transcoded. The size of the original speech payload remained the same, and the changes brought by the encoding were unpredictable. Conversely, after the audio payload was transcoded, the remaining unused space was filled with hidden data. The steganographic bandwidth obtained was 32 kbits/s with delays lower than 1 ms; however, the speech quality was not experimentally assessed using the standard method. In 2012, Janicki et al. (Janicki et al., 2012a) further extended the method and analysed the impact of the choice of the TranSteg speech codec. They argued that

TranSteg with the G.711/G.711.0 codec did not incur steganographic costs and provided a remarkably high steganographic bandwidth of 31 kbps on average.

The LSB methods in different protocols timestamp fields have different steganographic bandwidths. In 2012, Tian et al. (Tian et al., 2012) experimentally evaluated the steganographic bandwidth and undetectability of the two schemes proposed by Huang et al. (Huang et al., 2011a) (LSBs of NTP timestamp field of RTCP protocol) and by Forbes (Forbes, 2009) (LSBs of timestamp field of RTP protocol). The authors used the Windows Live Messenger voice conversations system and confirmed that using the first method the steganographic bandwidth available was 335 bit/s, while the second steganographic bandwidth was 5.1 bit/s. The latter method was more difficult to detect.

C. Hybrid Methods

Hybrid methods modify both the content of PDU's specific fields and UDP's time relationship. In 2008, Mazurczyk and Szczypiorski (Mazurczyk et al., 2008b) introduced a novel method called LACK (Lost Audio Packets Steganography), and it was later described and analysed (Mazurczyk et al., 2010). LACK relied on modifying the content of RTP packets and their time dependencies. This approach took advantage of RTP that an excessively delayed packet was not used by the receiver to receive the reconstruction of the data; that is, the packet was considered useless and therefore discarded. Therefore, covert communication was possible by introducing an intentional delay to select an RTP packet and using the hidden information to replace the original payload. In 2012, Mazurczyk (Mazurczyk, 2012) made a practical evaluation of the LACK-based prototype and studied the impact of the method on

voice transmission quality, which is arguable and impractical. In 2011, Hamdaqa and Tahvildari (Hamdaqa & Tahvildari, 2011) further extended the concept of LACK, providing reliability and fault tolerance mechanisms based on the Lagrange threshold. The complexity of steganographic analysis increased with the loss of partial steganographic bandwidth.

In 2012, Arackaparambil et al. (Arackaparambil et al., 2012) presented a simple VoIP steganography method in which chosen RTP packets were replaced with the secret information, and deliberately changed the header sequence number and / or timestamp of the RTP to make the packets look like over delay by the network. This scheme can be regarded as a variation of the LACK method described above.

In summary, a great deal of research has been conducted on the design of steganographic algorithms for covert VoIP communication, but few studies have been carried out to investigate the security of covert steganographic VoIP communication, such as the security of keys and key distribution in the steganographic system.

3.8 Summary

This chapter focuses on the key concerns and considerations of steganography. Steganography, together with encryption, aims to attain integrity, confidentiality and availability of computer and network systems. Authentication and key distribution make the exchange of data more secure.

Steganography is the act of concealed communication by hiding secret data in seemingly innocuous objects, aiming to make the very existence of embedded data a

secret. According to different classification criteria, the classifications of steganography can be varying. Considering all classification methods, the most widely used steganographic method is LSB embedding in images, which is a blind substitution based method of embedding.

Since a steganographic scheme is considered broken when the existence of the hidden message is detected, the statistical undetectability of the embedded data is one of the most important properties for covert communications over streaming media. Other properties such as imperceptibility, security, capacity and robustness are also vital evaluation criteria of covert communications. However, enhancing a certain requirement may negatively affect others. As a result, there should be compromise among these requirements.

The attacks on steganographic communication include passive, active, and malicious. The requirements for steganalysis can vary significantly depending on the operational scenarios.

CHAPTER 4 Covert VoIP Communications

This chapter covers a new information theoretic model for covert VoIP communications, novel steganographic algorithms, a new covert VoIP communication system, experimental set-up and evaluation criteria.

4.1 Covert Communications

Covert Communication is the secret communication between two or more parties where adversaries should not know that the communication is taking place (Cox et al., 2008). Streaming media communication such as VoIP is suitable for covert communications. VoIP communication is one of the most popular real-time services on the Internet. Since the Internet allows VoIP to provide low-cost, high-reliability and global services, VoIP has more advantages than traditional telephony.

VoIP streams often have a highly redundant representation, which permits the addition of a significantly large amount of secret data by means of simple and subtle modifications that preserve the perceptual content of the underlying cover object. With the increasing proportion of VoIP streams in Internet traffic, VoIP has become a better cover object for steganography than the static carriers such as text files, image files and audio files. In addition, VoIP connection is usually relatively short, so it is unlikely for attackers to detect the hidden data within VoIP streams. Their real-time

characteristics can be used to improve the security of the hidden data embedded in "dynamic" VoIP streams.

A great deal of research has been conducted for covert steganographic communications over streaming media. Pazarci et al. first reported a method of embedding data in scrambled MPEG video using scrambling operation together with the data embedding process prior to MPEG encoding (Pazarci & Dipcin, 2003). Dittmann et al. studied VoIP steganography and decryption techniques and suggested their algorithms (Dittmann, Hesse & Hillert, 2005). Kratzer et al. studied the information hiding algorithm for VoIP streaming media and designed the architecture and synchronisation mechanism of VoIP information hiding software (Kratzer et al., 2006). Aoki developed a lossless steganographic technique for G.711 telephony speech, and the embedding capacity of the method depended on the number of "0" in the speech sample (Aoki, 2008), so the practical application is very limited. Liu et al. analysed the parameters of G.729 coded speech frames, and identified the parameters and effective bits of G.729 speech coding which could be used for information hiding (Liu et al., 2008). Yu et al. designed a simple VoIP information hiding scheme, and the validity needs to be further confirmed (Yu et al., 2009). Aoki proposed a semi-lossless steganographic technique for G.711 telephony speech, which improved the bandwidth from 24 bit/s to 400 bit/s, but it depended on the background noise signal level (Aoki, 2010). Tian et al. developed a method to improve the performance of data hiding by adding some similarity between the hidden message and the cover object to achieve a balance between hiding transparency and bandwidth, but this similarity limited the choice of hidden messages (Tian et al., 2012).

Mazurczyk et al. suggested a transcoding information hiding method, which improved the information hiding capacity (32 Kbit/s) by compressing streaming media carrier packets (Mazurczyk et al., 2014). Tian et al. improved the security of quantization-index-modulation steganography in low bit-rate speech streams (Tian et al., 2014). In 2016, Qi et al. used Discrete Spring Transform (DST) to eliminate the perceived redundancy in multimedia signals in order to improve the quality of speech in steganography (Qi et al., 2016). Liu et al. reported the use of matrix embedding (ME) method to achieve information hiding in the linear predictive coding for low bit-rate speech codec (Liu et al., 2016). Janicki investigated pitch-based steganography using Speex voice codec, and its practicality needs further study (Janicki, 2016).

In recent years, Lin et al. proposed a novel data hiding algorithm for high dynamic range (HDR) images encoded by the OpenEXR file format, offering a high embedding rate and high visual quality of the stego image (Lin et al., 2017). Jiang et al. designed a reversible data hiding scheme in encrypted domain with low computational complexity for three-dimensional meshes (Jiang et al., 2018). Zhang et al. suggested a coverless image steganographic algorithm based on discrete cosine transform and latent dirichlet allocation (LDA) topic classification, having robustness against common image processing and better ability to resist steganalysis (Zhang et al., 2018). Yi et al. studied separable and reversible data hiding in encrypted images using parametric binary tree labelling, achieving an average embedding rate up to 2.003 bpp (Yi et al., 2019). More recently, Zhou et al. reported a distortion design for secure adaptive 3D mesh steganography, which relied on some effective steganalytic

features such as variation of vertex normal (Zhou et al., 2019). Overall, these steganography studies mainly focus on steganographic algorithm design. There is no existing information theoretical model that can be put into use for covert communications over streaming media. In this chapter, a novel information theoretical model of steganographic VoIP communication is constructed to realise secure covert VoIP communications, aiming to achieve high data embedding capacities comparable to other related algorithms.

4.2 Information-Theoretic Model of Covert Communications

4.2.1 Cachin's Definition of Steganographic Security

Shannon first propounded the communications theory of secrecy systems from the viewpoint of information theory and identified three general forms of secret systems: (1) concealment systems; (2) privacy systems; (3) cryptographic systems (Shannon, 1949). However, he declared that "concealment systems are primarily a psychological problem" and did not consider it further nor provide the precise definition of steganographic security. Cachin defined the steganographic security which is most widely used and first proposed an information-theoretic model for steganography with a passive adversary (Cachin, 1998).

Cachin's model is based on Simmons's "Prisoners' Problem", in which the adversary's decision of distinguishing between an innocent cover message and a modified message containing hidden message is regarded as a statistical hypothesis testing problem. The distances between the distributions of the cover object and the

stego object is quantified using the relative entropy. It also defines conditions for both perfectly secure and ε-secure steganographic systems. The model of a secret-key stegosystem is shown in Figure 4.1.

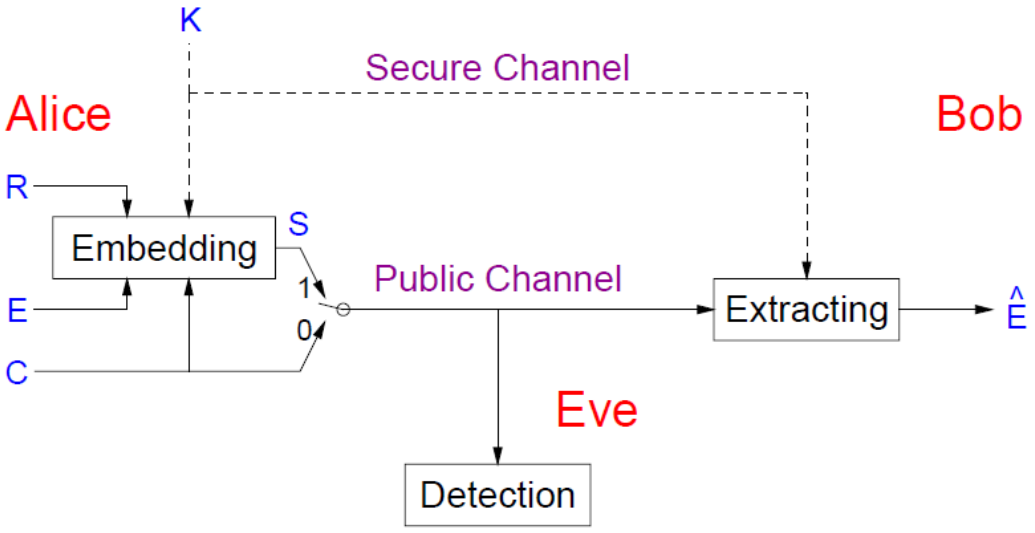

Figure 4. 1 Cachin's model of a secret-key stegosystem

According to Cachin's definition, when the relative entropy between the distributions of the covertext and stegotext $D(P_C \| P_S) = 0$, the stegosystem is regarded as perfectly secure against a passive adversary. If $D(P_C \| P_S) \leq \varepsilon$, the stegosystem is called ε-secure (Cachin, 1998). C and S in this definition denote covertext and stegotext respectively, P_C and P_S are the probability distributions of the covertext C and the stegotext S, and Q is the space of possible measurements.

$$D(P_C \| P_S) = \sum_{q \in Q} P_C(q) \log \frac{P_C}{P_S} \qquad (4.1)$$

Eve, the warden in Cachin's model can make two types of error and her decision performance uses the theory of hypothesis testing. If Eve decides that a stegotext is present when it is absent, the first form of error (type I) occurs. α denotes the

probability of this kind of error. The opposite error (type II) is that Eve fails to detect the stegotext *S,* and its probability is denoted by β. $d(\alpha,\beta)$ denotes the relative entropy between both distributions of Eve's detection.

$$d(\alpha,\beta) = \alpha \log \frac{\alpha}{1-\beta} + (1-\alpha)\log \frac{1-\alpha}{\beta} \qquad (4.2)$$

According to a standard result in information theory, when P_{Q_0} and P_{Q_1} are two plausible probability distributions over the space of possible measurements *Q*. For any function $f: Q \to \mathrm{T}$, if $T_0 = f(Q_0)$ and $T_1 = f(Q_1)$, then $D(P_{T_0} \| P_{T_1}) \le D(P_{Q_0} \| P_{Q_1})$ (Thomas, 1991).

Similarly,

$$d(\alpha,\beta) \le D(P_C \| P_S) \qquad (4.3)$$

In a ε-secure stegosystem, $D(P_C \| P_S) \le \varepsilon$, then $d(\alpha,\beta) \le D(P_C \| P_S) \le \varepsilon$, the probability β and the probability α satisfy $d(\alpha,\beta) \le \varepsilon$. In particular, when the probability of the type I error α = 0, then $\beta \ge 2^{-\varepsilon}$. Thus, the smaller ε is, the closer the two distributions P_C and P_S are.

In conclusion, since the physical meaning of ε describes the statistical distance between the cover object and the stego message, it can be used to quantify the security of the stegosystem. The smaller ε is, the greater the probability that covert communication is detected. When ε = 0 and β = 1, it cannot be detected by the adversary; in this case, the stegosystem can be called perfectly secure.

However, there are some limitations on Cachin's model. This theoretical model is mainly designed for text / image steganography and it is not suitable for real-time

covert communications over streaming media. Firstly, as Cachin's model assumed, the stego object is encrypted by a simple key only, this conventional model cannot manage the series of keys in the continuous process of embedding in streaming media. Secondly, due to packet loss, it is one of the major problems to synchronise between embedding and extraction in a real-time covert communication. But Cachin's model is short of synchronisation mechanisms. In addition, the model cannot address the embedding capacity for steganography in streaming media. Thus, a new information theoretical model for steganography in streaming media is needed.

4.2.2 Proposed Model of Secure Covert VoIP Communications

As stated in 4.2.1, Cachin's model is not suitable for steganography in streaming media. It deals with static cover objects into which the secret data is embedded statically, whereas streaming media are used for dynamic embedding in a real-time manner. In this study, a new theoretical model of covert steganographic communications over streaming media was devised, which is based on steganography and cryptography. The proposed model is applicable to covert VoIP communications, and is proved to be a ε- secure stegosystem below.

The model is depicted in Figure 4.2. As the figure displays, the secret message to be hidden (M) is encrypted using a secret key generated from a true random number to form an encrypted message, which is segmented into distinct parts that are then embedded in a series of packets of media streams, namely cover objects or works (c). C and S in the figure denote the packets without and with a hidden message,

respectively.

As Fig. 4.2 shows, the random vector, $A = \{a_1, a_2, \ldots a_n\}$, is a group of zeros, ones, twos and threes, e.g. {1,2,2,3,0,…}, describing the continuity of the data embedding process; ones, twos and threes denote a packet containing the beginning, the continuation and the end of a hidden message, respectively, and zeros mean a packet contains no hidden message.

The vector, $B = \{b_1, b_2, \ldots b_n\}$, is a set of steganographic capacity, corresponding to varying numbers of bits of the hidden message embedded in a series of packets of streaming media. This sequence enables the receiver to determine the size of the secret message embedded in each packet.

The vector, $L = \{l_1, l_2, \ldots l_n\}$, represents a set of private / public key pairs of 1024bits each, since the public key and the private key are correlated in the public-key scheme where the sender and the receiver calculate the shared private key using knowledge of the public key based on the discrete exponential and logarithm (hash) functions. The covert channels detailed below are used to conduct key transmission and key updating for covert steganographic communications over streaming media.

Assuming that the Sender and the Receiver share three sequences (set of variables), $A = a_0 a_1 a_2 \ldots, B = b_0 b_1 b_2 \ldots$and $L = l_0 l_1 l_2 \ldots$, respectively. The sequence *A* and *B*are an *N*-level *M* pseudo-random vectors with the period of $(2^n - 1)$ respectively; and in one period the probability of a fixed result is $\eta = [2^{n-1} / (2^n - 1)]$ ^ 2. The sequence B is used as the secret key for encryption.

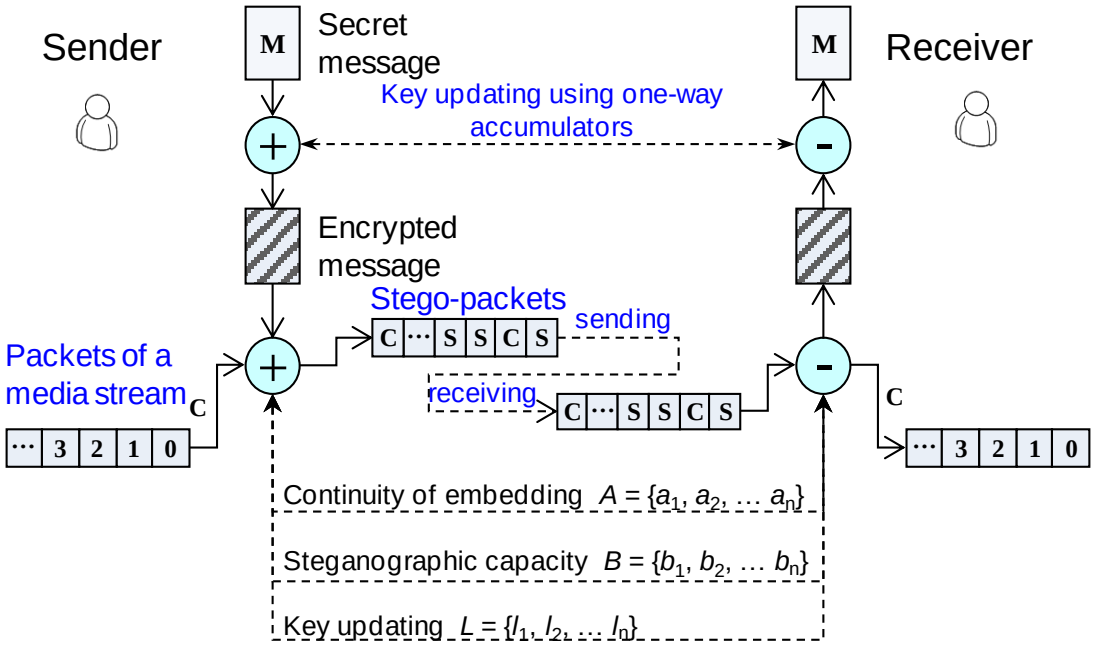

Figure 4. 2 Model of covert communications over streaming media

The statistical distance between the cover object and the secret message (the stego object) can be presented as $\varepsilon = \sum_{w \in Q_0} P_C(w) - \sum_{w \notin Q_0} P_C(w)$, $Q_0 \subset Q$, where P_C is the probability distribution of the cover object, Q_0 is a plausible space, and Q is the total space of possible measurements. Assuming the Sender and the Receiver share $Q_2 \subset Q$, let $\sum_{w \in Q_2} P_C(w) = \eta, Q_3 = Q - Q_2$, where Q_2 and Q_3 are the observation spaces in relation to the stego and cover object respectively. An information source is often appropriate to be modelled as a stochastic process U. If the secret message sent in the ith packet is denoted by U_t, the stochastic process $\{U_t , t = 0,1,2, \ldots\}$ represents the stego-packet S or the cover-packet C. Then the total probability distribution of the secret message sent over the space Q is given by

$$P_U(w) = P(T \in Q_2)P_S(w) + P(T \in Q_3)P_C(w) = P(a_i \in Q_2)P_S(w) + P(a_i \in Q_3)P_C(w) \tag{4.4}$$

Then

$$P_U(w) = \eta P_S(w) + (1-\eta) P_C(w) \tag{4.5}$$

As

$$P_S(w) = \begin{cases} P_C(w)/1+\varepsilon, & w \in Q_0 \\ P_C(w)/1-\varepsilon, & w \in Q_1 \end{cases} \tag{4.6}$$

Where Q_0 and Q_1 are two plausible spaces of possible measurements, then

$$P_U(w) = \begin{cases} \eta \dfrac{P_C(w)}{1+\varepsilon} + (1-\eta) P_C(w) = P_C(w) \dfrac{1+\varepsilon(1-\eta)}{1+\varepsilon}, & w \in Q_0 \\ \eta \dfrac{P_C(w)}{1-\varepsilon} + (1-\eta) P_C(w) = P_C(w) \dfrac{1-\varepsilon(1-\eta)}{1-\varepsilon}, & w \in Q_1 \end{cases} \tag{4.7}$$

According to information theory, the relative entropy between the cover object and the stego object for steganography in streaming media is given by

$$\begin{aligned} D(P_C \,||\, P_U) &= \sum_{w \in Q} P_C(w) \log \frac{P_C(w)}{P_U(w)} \\ &= \sum_{w \in Q_0} P_C(w) \log \frac{1+\varepsilon}{1+\varepsilon(1-\eta)} + \sum_{w \in Q_1} P_C(w) \log \frac{1-\varepsilon}{1-\varepsilon(1-\eta)} \\ &= \frac{1+\varepsilon}{2} \log \left[\frac{1+\varepsilon}{1+\varepsilon(1-\eta)} \right] + \frac{1-\varepsilon}{2} \log \left[\frac{1-\varepsilon}{1-\varepsilon(1-\eta)} \right] \\ &\le \frac{1+\varepsilon}{2} \left[\frac{\varepsilon\eta}{1+\varepsilon(1-\eta)} \right] + \frac{1-\varepsilon}{2} \left[\frac{-\varepsilon\eta}{1-\varepsilon(1-\eta)} \right] = \frac{\varepsilon^2 \eta^2}{1-\varepsilon^2(1-\eta)^2} \end{aligned} \tag{4.8}$$

For the *N*-level *M* pseudo-random sequences, $\eta = [2^{n-1} / (2^n - 1)]$ ^ 2 = 1/4,

$$D(P_C \,||\, P_U) \le \varepsilon^2 / (16 - 9\varepsilon^2) \tag{4.9}$$

The relative entropy $D(P_C \,||\, P_U)$ does satisfy the condition defined in Cachin's theory, i.e.

$$D(P_C \,||\, P_U) \le \varepsilon^2 / (16 - 9\varepsilon^2) \le \varepsilon \tag{4.10}$$

Therefore, it proves the proposed model for steganography in streaming media is ε-

secure against a passive adversary.

The proposed model of secure covert VoIP communications integrates dynamic key updating with the data embedding and extraction processes, thereby realising secure covert communications over VoIP streaming media with a greater data embedding rate.

4.3 Covert VoIP Communications Algorithm

The basic process of covert VoIP communications is the combination of steganography and cryptography. Firstly, it generates keys used for encrypting the secret data and distributes the keys. Secondly, the secret message to be hidden is encrypted using the secret key generated from a true random number to form an encrypted message. Then, the embedding and extraction algorithms are used in the process of VoIP communications to realise covert communications. Thus, a covert VoIP communication system includes VoIP communication, key generation, key exchange and updating, encryption, data embedding, data extraction and decryption.

4.3.1 Advanced Encryption

Advanced Encryption Standard (AES) is a specification of the standard encryption established by the U.S. National Institute of Standards and Technology (NIST), which is based on Rigndael cipher algorithm (Daemen & Rijmen, 2003). The computational complexity of breaking the AES-128 encryption is $2^{126.1}$, which indicates that the attacks to this encryption algorithm are computationally infeasible. In addition, the AES-128 is one of the fastest algorithms for encryption, which can meet the real-time

requirements of VoIP communications. Therefore, this encryption algorithm was used to encrypt the secret data in the covert VoIP communication system developed in this project.AES is a symmetric block cipher. In an AES-128 encryption algorithm, the block size of the plaintext and the cipher text are 16 bytes, or expressed as 128 bits, and the key length is 128 bits (16 bytes) (FIPS 197, 2001). In this algorithm, the cipher consists of 10 rounds. The first 9 rounds consist of four different byte-oriented transformation functions: byte substitution (SubBytes ()), shifting rows (ShiftRows ()), mixing data in each column (MixColumns ()), and adding RoundKey (AddRoundKey ()). The last round only contains three transformations and there is a single transformation function (AddRoundKey ()) to initialise before the first round. Figure 4.3 shows the partial pseudo code of AES encryption in which Nb is the block size (words) and Nr is the number of rounds (words) (FIPS 197,2001).

```
Cipher algorithm
Input: byte in [4*Nb],byte out [4*Nb],word w[Nb*(Nr+1)])
    begin
        byte state[4, Nb]
        state = in
        AddRoundKey(state, w[0, Nb-1])
        for round=1 step 1 to Nr-1
            SubBytes(state)
            ShiftRows(state)
            MixCloumns(state)
            AddRoundKey(state, w[Nr*Nb, (Nr+1)*Nb-1])
        end for
        SubBytes(state)
            ShiftRows(state)
            AddRoundKey(state,w[Nr*Nb, (Nr+1)*Nb-1])
            out= state
    end
```

Figure 4. 3 Partial pseudo code of the AES encryption

4.3.2 Data Embedding Algorithm

A data embedding algorithm was devised for the covert VoIP communications system developed in this project. In the system, AES-128 is used to encrypt the secret message before embedding it into VoIP streams to improve the secrecy. Since VoIP streams are transmitted on the public network, the impact of delay and packet loss is inevitable. Considering that the extraction and decryption of the secret message would be affected by packet loss, the encrypted message is segmented into distinct blocks before data embedding, so that the original secret message can be separately extracted and decrypted from independent VoIP voice packets.

It is necessary to shorten the time required to encrypt the secret message for real-time VoIP communication. AES-128 encryption is a block cipher which takes the plaintext block size of 128 bits (or 16 bytes), and the encryption time is less than 0.1ms. In order to embed the secret message as much as possible in the VoIP voice packet and reduce the encrypting time to meet the needs of real-time communication, the length of the secret message embedded in each single VoIP packet should be a multiple of 16 bytes and it must be smaller than the embedding capacity of a single voice packet.

The encrypted secret message is segmented and then embedded in the packets of VoIP streams at various embedding capacities and different embedding locations. In the covert VoIP communication system, the size of data in each VoIP packet is set to be 4096 bytes. The first 16 bytes of the first VoIP packet is used to embed the length of the secret message to be hidden (LoM), and the remaining of the first packet and the other VoIP packets are used to embed the secret message itself. The size of the

remaining of the first VoIP packet is 4080 bytes, and the data embedding capacity is up to 510 bytes (12.5%). To reduce the encryption time, the first 496 bytes of the secret message are embedded in the first VoIP packet as it is a multiple of 16 bytes. As for the other VoIP packets, 512 bytes of the encrypted secret message are embedded in each VoIP packet. The remaining of the secret message is embedded in the last VoIP packet with the size of LLoM. The LLoM value may not be a multiple of 16 bytes, so it is possibly necessary to adjust it to a multiple of 16 bytes in some cases (Peng et al., 2020).

In order to achieve variable embedding capacity, a parameter R is set as the data embedding interval to minimise the impact on voice quality in different environments and different lengths of the secret message (Peng et al., 2020). The larger the embedding interval is, the less the secret message could be embedded into each VoIP packet. For simplicity, 16 bytes of VoIP streams have a data embedding capacity of two bytes. When the value of R is set to be one, the interval of the secret message embedded is two bytes, *i.e.* one byte of the secret message is randomly embedded in 16 bytes of VoIP streams. If the interval is set to two, then the secret message is randomly embedded in VoIP streams at an interval of three bytes, and in this case 22 bytes of VoIP streams contain one byte of the secret message, and so on.

The process of embedding the secret message in VoIP streams is designed as follows:

Step A: first embed the secret message length (LoM) in VoIP streams, and set length to LoM.

Step B: compute the length of the secret message hidden in the first packet (m_1) and the length of the secret message hidden in other packets (m_k). If the length is less than the message size in the first packet, encrypt $M(0, m_0 - 1)$ to form $E(0, m_0 - 1)$, and then embed $E(0, m_0 - 1)$ in the bit stream $BIT = \{bit(0), bit(1), \ldots, bit((m_0 - 1) * 8)\}$.

If the bit stream is equal to zero

Set vector $V(k)$ to $V(k)$&0xfe

If the bit stream is equal to one

Set vector $V(k)$ to $V(k)$|0x01

Add R into k

Set the length to zero

End

where m_0 is the total size of the secret message to be hidden, and k is the sequence number. m_1 denotes the length of the secret message embedded into the first packet, m_k represents the length of the secret message embedded into other packets, m_n denotes the length of the secret message in the last packet. & is a bitwise AND operation, and | is a bitwise OR operation. The prefix 0x is used in C and related programming languages; 0xfe is a hexadecimal value and the decimal value is 254; 0x01 is the least significant bit set and the decimal value is 1.

If the length is greater than the message size in the first packet, encrypt $M(0, m_1 - 1)$ to form $E(0, m_1 - 1)$, and embed $E(0, m_1 - 1)$ in the bit stream $BIT = \{bit(0), bit(1), \ldots, bit((m_1 - 1) * 8)\}$.

If the bit stream is equal to zero

Set vector *V*(k) to *V*(k)&0xfe

If the bit stream is equal to one

Set vector *V*(k) to *V*(k)|0x01

Add *R* into k

Reduce the length by the secret message size in the first packet

Step C: if the length is greater than the message length in other packets, encrypt $M(m_1, m_1 + m_k - 1)$ to form $E(m_1, m_1 + m_k - 1)$, and embed $E(m_1, m_1 + m_k - 1)$ in the bit stream $BIT = \{bit(0), bit(1), \ldots, bit((m_k - 1) * 8)\}$.

If the bit stream is equal to zero

Set vector *V*(k) to *V*(k)&0xfe

If the bit stream is equal to one

Set vector *V*(k) to *V*(k)|0x01

Add *R* into k

Reduce the length by the secret message size in other packets

Repeat Step C until the length is less than the secret message size in other packets

Step D: compute the length of the secret message hidden in the last packet (m_n), encrypt $M(\text{LoM}- \text{length}, \text{LoM}- 1)$ to form $E(\text{LoM}- \text{length}, \text{LoM}- \text{length} + m_n - 1)$, and embed $E(\text{LoM}-\text{length}, \text{LoM}-\text{length}+ m_n -1)$ in the bit stream $BIT = \{bit(0), bit(1), \ldots, bit((m_n - 1) * 8)\}$.

If the bit stream is equal to zero

Set vector *V*(k) to *V*(k)&0xfe

If the bit stream is equal to one

Set vector *V*(k) to *V*(k)|0x01

Add *R* into k

Set the length to zero

End

Data processing methods could be used in the process of data embedding to improve the secrecy of the covert VoIP system, such as using a self-adaptive method to detect whether a VoIP packet carries active or inactive audio data and using a logistic map to choose the embedding location randomly (Peng et al., 2020).

4.3.3 Data Extraction Algorithm

The extraction of the secret message, steganographically embedded in VoIP streams using the data embedding algorithm above, from the stego VoIP streams is the inverse process of the data embedding algorithm. The corresponding extraction algorithm is used to retrieve the secret message encrypted with AES, and decrypt it with the same secret keys to obtain the original secret message from stego VoIP packets.

4.4 Development of Covert VoIP Communication System

Based on the proposed theoretical model, a covert VoIP communication system called StegPhone was developed in this project to perform covert communication experiments. In this system, end-user terminals are connected to a VoIP proxy server on an IP local network.

The communication parties establish connections by SIP signalling protocol, the real-time audio communications are based on the media protocol, and the IP protocol handles the VoIP voice transmission. All these protocols were implemented on the VoIP server which is connected to end-user terminals.

In the covert VoIP communication system, the sender saves the secret message to be hidden into a .txt file, and then choose the file in the system to embed the secret message into the cover object. After receiving and extraction, the receiver obtains a .txt file which contains the secret message.

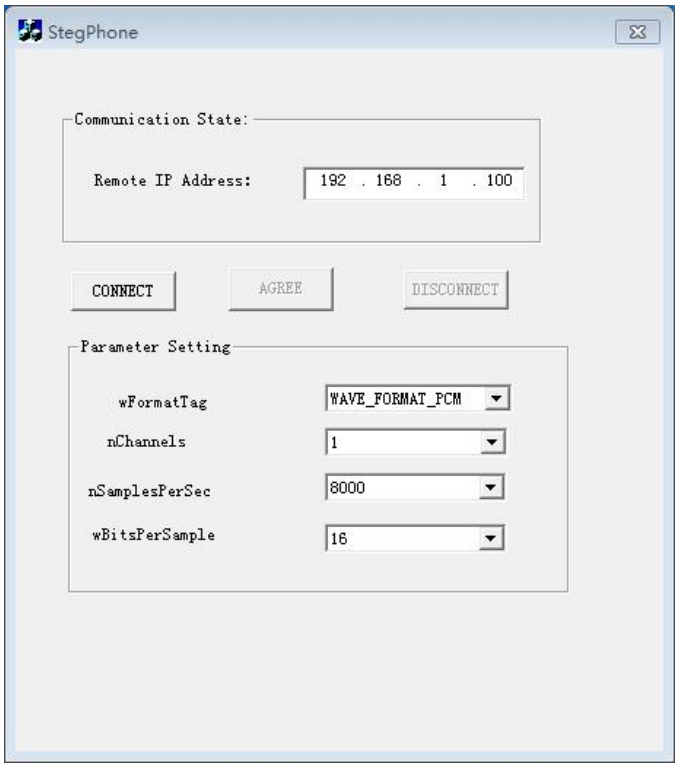

Figure 4. 4 End-user interface of the covert VoIP communication system

The covert communication system was developed using C++ and MFC. The speech signal acquisition and playback is implemented by means of winmm.lib which is a multimedia API, and the real-time transmission of streaming media audio packets is based on jrtplib 3.9.1 library. The system uses the User Datagram Protocol (UDP)

rather than Transmission Control Protocol (TCP) to establishing low-latency and loss-tolerating connections between VoIP communicating parties. Figure 4.4 shows the interface of end-user.

The end-user's IP address is needed to initialise covert VoIP communication. The parameters used for encoding, sampling and quantizing for VoIP streams are then set. In the covert communication system, VoIP audio samples are coded by PCM using single-channel and sampling at 8000 Hz, and are used as cover objects. Each speech sample is represented with 16 bits. As Figure 4.4 shows, there are three buttons that can be used to control the connection between the communicating parties. The "CONNECT" button on the sender side is used to initiate a session, and by clicking the "AGREE" button on the receiver side the two end-users are connected and the communication starts. The "DISCONNECT" button is used to terminate the connection.

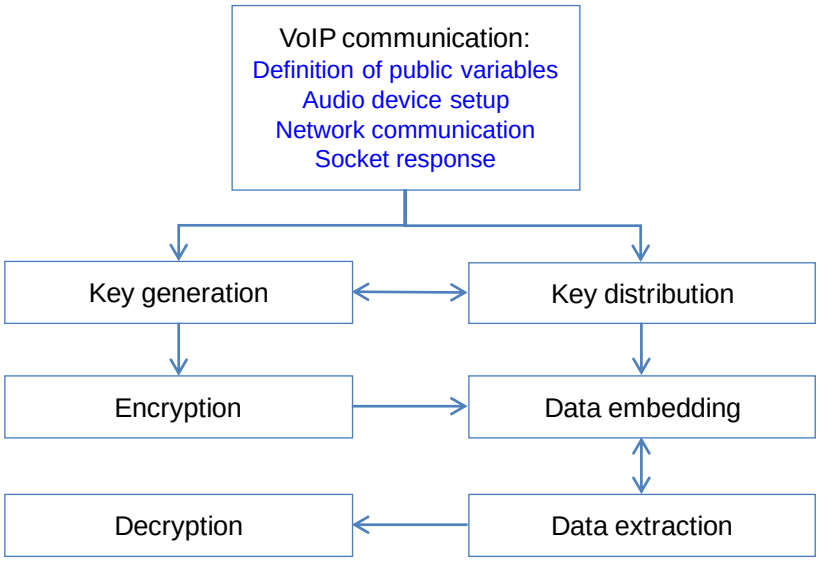

Figure 4. 5 Diagram of a covert VoIP communication system

The covert communication system is capable of addressing the key distribution and

generation of true random keys over streaming media. It consists of several main modules, including VoIP Communication, Random Key Generation, Key Distribution, Encryption, Data Embedding, Data Extraction and Decryption and so on. Figure 4.5 shows how the modules of a covert VoIP communication system interact to perform covert communication.

4.4.1 VoIP Communications Module

The main parts in the VoIP communications module include definition of public variables, audio device set up, network communications, socket response and so on.

A. Definition of Public Variables

Public variables are defined in the source code NetPhoneDlg.cpp, such as the buffer size and handle. Figure 4.6 shows the detail of public variables definition.

```
#ifdef _DEBUG                          intnComState=1;
#define new DEBUG_NEW                  BOOL bDisconnectState = TRUE;
#undef THIS_FILE                       BOOL bBtnConnectDown = FALSE;
static char THIS_FILE[] = __FILE__;    BOOL bServerState = FALSE;
#endif                                 BOOL bClientState = FALSE;


#define INP_BUFFER_SIZE 4096           CSocketServerSocket_Server;
#define HIDE_SIZE 10000                CSocketClientSocket_Client;
#define WM_NC 1001
#define IDC_NC 1002                    intSockaddrinLength;
                                       intSockaddrLength;
static HWAVEIN hWaveIn;                UINT inport;
static HWAVEOUT hWaveOut;
static PBYTE pBufferIn[2];             CStringLocalIP;
static PBYTE pBufferOut[2];            CStringsRemoteIP;
static PWAVEHDR pWaveHdrIn[2];         SOCKADDR_IN LocalAddr;
static PWAVEHDR pWaveHdrOut[2];        SOCKADDR_IN RemoteAddr;
static WAVEFORMATEX waveform;
                                       CStringsAck;
intnIn=0;                              charcAck[15];
intnOut=0;
```

Figure 4. 6 Definition of public variable

B. Audio Device Set Up

```
voidCNetPhoneDlg::RecordBegin()
{
        waveInAddBuffer(hWaveIn,pWaveHdrIn[nOut],sizeof(WAVEHDR));
        waveInStart(hWaveIn);
}

voidCNetPhoneDlg::ON_MM_WIM_OPEN()
{
}

LRESULT CNetPhoneDlg::ON_MM_WIM_DATA(WPARAM wParam, LPARAM lParam)
{
        Socket_Client.SendTo(pBufferOut[nOut],INP_BUFFER_SIZE,(SOCKADDR*)&RemoteAddr,SockaddrLength);
        nOut=1-nOut;
        if(bDisconnectState == TRUE) waveInReset(hWaveIn);
        elseRecordBegin();
        return 0;
}

LRESULT CNetPhoneDlg::ON_MM_WIM_CLOSE(WPARAM wParam, LPARAM lParam)
{
        UNREFERENCED_PARAMETER(wParam);
        UNREFERENCED_PARAMETER(lParam);
        waveInUnprepareHeader(hWaveIn,pWaveHdrIn[0],sizeof(WAVEHDR));
        waveInUnprepareHeader(hWaveIn,pWaveHdrIn[1],sizeof(WAVEHDR));
        waveInClose(hWaveIn);
        hWaveIn=NULL;
        return 0;
}
  LRESULT CNetPhoneDlg::ON_MM_WOM_OPEN(WPARAM wParam, LPARAM lParam)
{
Socket_Server.ReceiveFrom((void*)pBufferIn[nIn],INP_BUFFER_SIZE,sRemoteIP,inport);
        waveOutWrite(hWaveOut,pWaveHdrOut[nIn],sizeof(WAVEHDR));
        nIn=1-nIn;
        if(nComState%10 == 1)
                ::SetDlgItemText(pDlg->m_hWnd,IDC_STATIC_INFORMATION,"In-call");
        else
                ::SetDlgItemText(pDlg->m_hWnd,IDC_STATIC_INFORMATION," ");
        nComState++;
        return 0;
}
```

Figure 4. 7 Partial pseudo-code of the audio device set up

Several functions are used to set up audio devices used for VoIP communication. The

function *InitAudioDevice*() is defined to initialise the audio device. The function *RecordBegin*() is used for starting record. For the input device, the MM_WIM_OPEN message is sent to a window when a waveform-audio input device is opened. The MM_WIM_DATA message is sent to a window when waveform-audio data is present in the input buffer, and the MM_WIM_CLOSE message is sent to a window when a waveform-audio input device is closed. The MM_WOM_OPEN message is sent to a window when the given waveform-audio output device is opened. Figure 4.7 shows partial pseudo-code of the audio device set up.

C. Network Communications

The functions of CSocketServer() are implemented in the C++ source file SocketServer.cpp and the functions of CSocketClient() were implemented in SocketClient.cpp. CSocketServer member functions include *OnAccept*(intnErrorCode) and *OnReceive*(intnErrorCode). The member functions of CSocketClient include *OnConnect*(intnErrorCode) and *OnSend*(intnErrorCode).

D. Socket Response

In the source file NetPhoneDlg.cpp, the information response functions are implemented using a socket. The pseudo-code is shown in Figure 4.8.

```
voidCNetPhoneDlg::OnButtonConnect()
{
    pickstart = 3;
    UpdateData(TRUE);
    bBtnConnectDown = TRUE;
    bClientState = TRUE;
    GetDlgItemText(IDC_IPADDRESS1,m_sServerIP);
        int tag=0;
        tag=Socket_Client.SendTo("COMMUNICATION",13,(SOCKADDR*)&RemoteAddr,SockaddrLength);
        GetDlgItem(IDC_BUTTON_CONNECT)->EnableWindow(FALSE);
        GetDlgItem(IDC_BUTTON_DISCONNECT)->EnableWindow(TRUE);
}
voidCNetPhoneDlg::OnButtonCommunicate()
{
    pickstart = 2;
    GetDlgItem(IDC_BUTTON_CONNECT)->EnableWindow(FALSE);
    GetDlgItem(IDC_BUTTON_COMMUNICATE)->EnableWindow(FALSE);
    GetDlgItem(IDC_BUTTON_DISCONNECT)->EnableWindow(TRUE);
    bDisconnectState = FALSE;
        if(m_bFirstRunAudio == TRUE)
    {
            if(InitAudioDevice()) m_bFirstRunAudio = FALSE;
            else
            {
                    AfxMessageBox(_T("Initialization waveform audio equipment failed!"),MB_ICONINFORMATION |
MB_OK,NULL);
                    return;
            }
        }
    if(bClientState == FALSE)
            {
            Socket_Client.SendTo(sztext1,sizeof(sztext1),(SOCKADDR*)&RemoteAddr,SockaddrLength);
            int tag=0;
            tag = Socket_Client.SendTo("AGREE",5,(SOCKADDR*)&RemoteAddr,SockaddrLength);          if(tag==5)
                        {       SetDlgItemText(IDC_STATIC_INFORMATION,sRemoteIP);
                                bClientState=TRUE;
                                bServerState=TRUE;
                                RecordBegin();
                        }
                }
                elseRecordBegin();
            }
            voidCNetPhoneDlg::OnButtonDisconnect()
```

```
{
    if(In_dataBuf !=NULL)   SaveInFile();
    if(m_dataBuf !=NULL)      SaveFile();
    if(bBtnConnectDown ==FALSE)
        Socket_Client.SendTo("NO",15,(SOCKADDR*)&RemoteAddr,SockaddrLength);
    else
    {
        bDisconnectState = TRUE;
        bServerState = FALSE;
        bClientState = FALSE;
        bBtnConnectDown = FALSE;

        Socket_Server.ShutDown();
        Socket_Client.ShutDown();
        Socket_Server.Close();
        Socket_Client.Close();
```

Figure 4. 8 Partial pseudo-code of socket response

4.4.2 Key Generation and Distribution Module

A. Random Key Generation

An entropy source-based method that uses the Read Time Stamp Counter of the CPU was devised to generate true random numbers as dynamic keys for encryption and covert steganographic communications over streaming media. Figure 4.9 illustrates the true random number generator.

```
__int64 CDiffieHellman::GetRTSC( void )
{
        int tmp1 = 0;
        int tmp2 = 0;
        __asm
        {
                RDTSC;
                mov tmp1, eax;
                mov tmp2, edx;
        }
        return ((__int64)tmp1 * (__int64)tmp2);
}
unsigned __int64 CDiffieHellman::GenerateRandomNumber(void)
{
static unsigned long rnd = 0x41594c49;
        static unsigned long x      = 0x94c49514;
        LFSR(x);
        rnd^=GetRTSC()^x;
        ROT(rnd,7);
        return (unsigned __int64)GetRTSC() + rnd;
}
```

Figure 4. 9 Partial pseudo-code of true random number generation

B. Key Distribution

A novel dynamic key exchange and updating algorithm using a one-way accumulator was designed to improve the security of the covert VoIP communication system. Figure 4.10 shows the partial pseudo-code of key distribution.

```
intCDiffieHellman::CreateSenderInterKey(__int64 &InterKey)
{
        if ( GetIsPublicKeyCreated() == false )
                return -1;
        a = (__int64) (GenerateRandomNumber() %MAX_RANDOM_INTEGER);
        X = XpowYmodN(g,a,n);
        InterKey = X;
        return 0;
}
intCDiffieHellman::CreateRecipientInterKey(__int64 &InterKey, __int64 Generator, __int64 Modulus)
{
        b = (__int64) (GenerateRandomNumber() % MAX_RANDOM_INTEGER);
        g = Generator;
        n = Modulus;
        Y = XpowYmodN(g,b,n);
        InterKey = Y;
        return 0;
}
intCDiffieHellman::CreateSenderEncryptionKey(__int64 &EncryptionKey, __int64 RecipientInterKey)
{
        Y = RecipientInterKey;
        K = XpowYmodN(Y,a,n);
        EncryptionKey = K;
        return 0;
}
intCDiffieHellman::CreateRecipientEncryptionKey(__int64 &EncryptionKey, __int64 SendersInterKey)
{
        X = SendersInterKey;
        K = XpowYmodN(X,b,n);
        EncryptionKey = K;
        return 0;
}
```

Figure 4. 10 Partial pseudo-code of key distribution

4.4.3 Data Embedding and Extraction Module

A. Data Embedding

According to the data embedding algorithm stated in section 4.3.2, the secret message is segmented into parts and then embedded in the packets of VoIP streams.

Figure 4.11 shows the pseudo-code of data embedding. The data embedding process includes four steps as the figure shows. Firstly, it embeds the length of the secret message into VoIP media streams. Secondly, it calculates the length of the secret message hidden in different VoIP packets. Thirdly, it embeds the encrypted secret message into VoIP streams. Lastly, it computes the length of data embedded in the last packet and embeds into VoIP streams.

```
voidCNetPhoneDlg::Embed()
{
    if(first_voice == 0)
        {   int length=LoM;
            charlencs[16]={0};
            Int_To_Binary(length,lencs);
            for(inti=15;i>=0;i--)
            {
                if(lencs[i]==1)
                    *test = *test|0x01;
                if(lencs[i]==0)
                    *test = *test&0xfe;
                test++;
            }
            if(LoM<first_packet)
            {
                if(LoM%16) hide_num=(LoM/16+1)*16;
                elsehide_num=LoM;
            }
            elsehide_num=first_packet;
            first_voice=1;
        }
    if(hide_num==0)
    {
        if((LoM-count)<middle_packet)
        {
if(LoM%16) hide_num=((LoM-count)/16+1)*16;
elsehide_num=LoM-count;
```

```
        }
                elsehide_num=middle_packet;
        }
        aes.Cipher(text,hide_num);
while(hided<hide_num)
        {
                cs=binary_print(*(text++));
                for(int k=0;k<8;k++)
                {
                        if(cs[k]=='1')
                                *test = *test|0x01;
        if(cs[k]=='0')
                                *test = *test&0xfe;
                        test++;
                }
                hided++;
        }
        count=count+hided;
        return;
}

char* CNetPhoneDlg::binary_print(char c)
{
        char temp[8];
        for(inti = 0; i< 8; ++i)
        {
                if(c <<i& 0x80)
                temp[i]='1';
                else
                temp[i]='0';
        }
return temp;

}
```

Figure 4. 11 Partial pseudo-code of data embedding

B. Data Extraction

The extraction of the hidden message is the inverse process of the data embedding process, and is depicted in Fig. 4.12.

```
voidCNetPhoneDlg::Pick()
{
    if(getfirstvoice==1)
    {
        for(inti=0;i<16;i++)
        {
            if(*c & 0x01)
                LoMs+= pow((double)2,15-i);
            c++;
        }
        if(LoMs<first_packet)
        {
            if(LoMs%16)
            {
                Pick_Num = (LoMs/16+1)*16;
                correct=LoMs;
            }
            elsePick_Num = LoMs;
        }
        elsePick_Num = first_packet;
        getfirstvoice=0;
    }
        if(Pick_Num == 0)
    {
        if((LoMs-getcount)<middle_packet)
        {
            if((LoMs-getcount)%16)
            {

correct=LoMs-getcount;
                Pick_Num=((LoMs-getcount)/16+1)*16;
            }
            elsePick_Num = LoMs-getcount;
        }
        elsePick_Num=middle_packet;
    }

    while(Picked<Pick_Num)
    {
        for(inti=0;i<8;i++)
        {
            if(*c & 0x01) temp+=pow(2.,(double)(7-i));
            c++;
        }
        ac[Picked]=(char)temp;
        Picked++;
        temp=0;
    }
    getcount=getcount+Picked;
    aes.InvCipher(ac,Pick_Num);
    file3.Write(ac,Pick_Num);
```

Figure 4. 12 Partial pseudo-code of data extraction

4.5 Experimental Set-up and Evaluation Criteria

4.5.1 Performance Measurement

The covert VoIP communication experiments were designed to assess the security and effectiveness of the proposed steganographic algorithm at addressing the key distribution issue and real random keys with VoIP communications over streaming media. The performance measurements were carried out by using the state-of-the-art network equipment Digital Speech Level Analyser (DSLA).

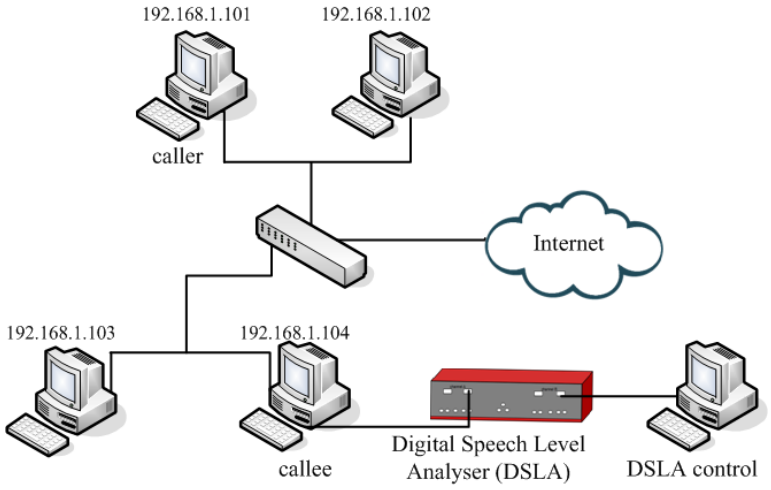

Figure 4. 13 Diagram of performance measurements for covert VoIP communications.

In the experiments, VoIP speech samples with PCM format encoded by G.711 codec were employed as cover objects for covert VoIP communications. The secret message to be embedded was encrypted using keys generated from the true random number generator, and then segmented into parts which were embedded into VoIP packets. The performance was evaluated by comparing the original cover object and the stego VoIP stream to assess the imperceptibility of the resulting stego VoIP streams, and the data embedding capacity was calculated according to each set of experimental results. DSLA was used to measure the SNR value and PESQ score of

the speech samples, which are two important parameters for performance measurements. Figure 4.13 shows a diagram of performance measurements for covert VoIP communications.

Digital Speech Level Analyser II, made by Malden Electronics Ltd, UK, was used to measure the performance of speech samples (DSLA, 2013). DSLAII provides a pair of stable, high quality analogue interfaces to a telephone network, which can be used to test a variety of telephone products, including telephone lines, telephone handsets, mobile phones and microphones or earpieces (DSLA, 2013). In the experiments, one DSLA is needed to carry out the local testing of the covert VoIP communication system. Figure 4.14 shows the experimental setup using DSLA II.

Figure 4. 14 Digital Speech Level Analyser (DSLA)

Figure 4.15 shows the internal components and the simplified block diagram of the DSLA II (DSLA, 2013). For input circuitry, there are two channels in the DSLA. From the user interface application, the DSLA application or MultiDSLA, and each channel can be switched to use different connector types, including Phoneline, Balanced and Handset (DSLA, 2013). When a connector type is selected the input is switched, whereas all outputs transmit the output signal. Isolation circuits and signal

conditioning are included between the connector terminals and the analogue/digital converters (ADC). The DSLA has 24-bit ADC and 24-bit DAC, which provide about 104 dB of dynamic range. From the user interface, the audio monitor can be configured to choose the intended monitor in terms of channel, input/output and level. As the figure shows, there are also battery backup memory, flash memory and digital signal processor (DSP) inside the DSLA. DSLA with bespoke software provides command, control functionality and real-time signal detection and processing (DSLA, 2013). The processing required to perform speech quality processing is divided between the DSP and the host processor of the controlling application.

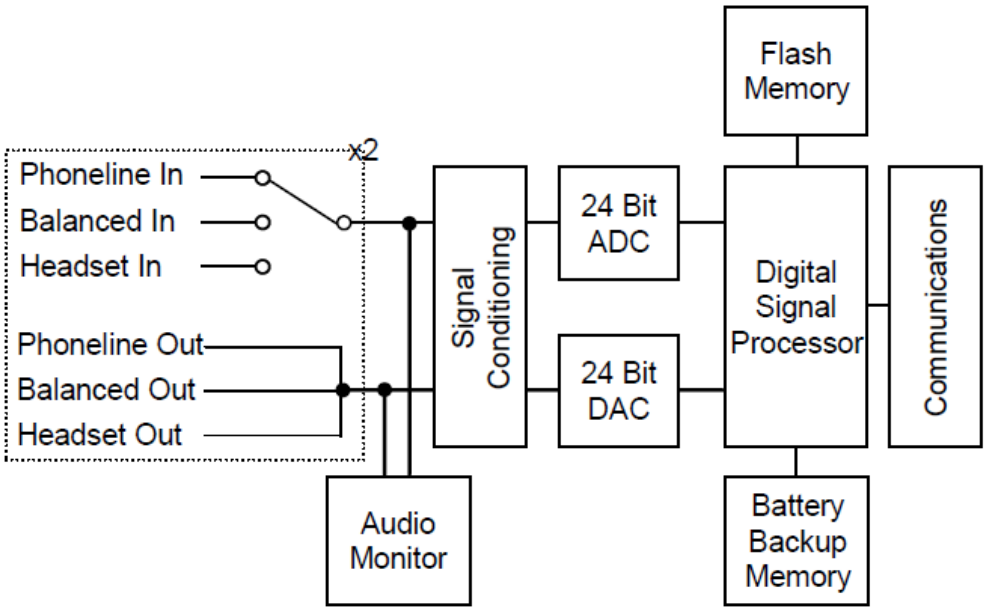

Figure 4. 15 Simplified block diagram of the DSLA

The DSLA and its user interface have been designed to provide access to the SNR value and PESQ score measurements, either directly from the analogue connection or from recorded speech files. It performs ITU-T P.862 objective speech quality scoring plus improved Mean Opinion Score prediction according to ITU-T P.862.1. This is a fully conformant implementation of PESQ as defined in ITU-T P.862, and can

be used to measure the performance of the steganographic algorithm.

4.5.2 Evaluation Criteria

The signal-to-noise ratio (SNR) is a measure which compares the level of a desired signal to a background noise, and it is one of the most commonly used measurements of speech quality in the field of streaming media steganography. In the experiments, larger SNR values mean better quality of the stego audio streams preserving the original audio streams. The SNR value can be calculated by using equations below:

$$SNR = 20\log(\frac{255}{\sqrt{MSE}}) \tag{4.11}$$

$$MSE = \frac{1}{N \times M}\sum_{i=0}^{N-1}\sum_{j=1}^{M-1}\left(\begin{matrix} PV\ of\ \text{cover}(i,j) \\ -PV\ of\ \text{stego}(i,j)\end{matrix}\right)^2 \tag{4.12}$$

where PV presents the pixel value, and the mean squared error (MSE) is a square value of the difference between the pixel value of cover audio streams and the pixel value of stego VoIP audio streams. Through the aforementioned formulae, it can be seen that the SNR value is inversely proportional to the MSE value. If MSE is equal to zero, SNR becomes infinite sequentially, meaning that no distortion occurs after data embedding (Peng et al., 2020).

Modern communication systems cannot be reliably assessed by conventional engineering metrics such as SNR alone. One solution to measuring customers' perception of the quality of the systems is to conduct subjective tests involving panels of human subjects. However, these types of tests are expensive and unsuitable for real-time monitoring applications. Perceptual evaluation of speech quality (PESQ)

provides an objective measurement which predicts the results of subjective listening tests on audio communication systems. To measure the quality of speech, PESQ uses a sensory model to compare the original signal with the degraded signal of the communication system. This process is shown in Figure 4.16.

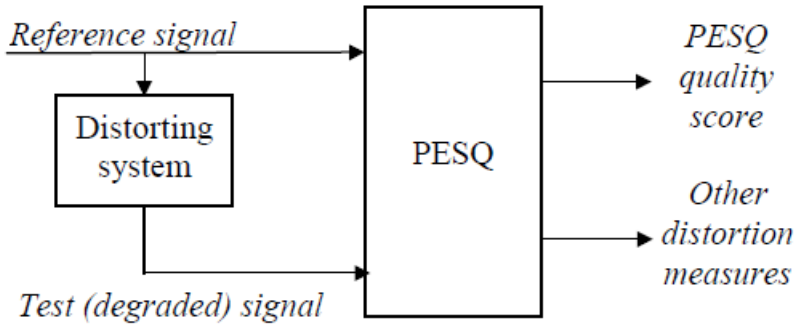

Figure 4. 16 Signal process in PESQ

The result of comparing the reference signal and the degraded signals is a quality score. This score is analogous to the subjective “Mean Opinion Score” (MOS) measured using panel tests according to ITU-T P.800. PESQ takes into account several sources of signal degradation, including coding distortions, packet loss, transmission errors, delay and variable delay, and the filtering in analogue network components. PESQ has not considered the subjective effect of level changes in the network, echo and the effect of round-trip delay on conversation (Malden Electronics Ltd, 2007).

PESQ requires two speech signals as inputs: the original test signal and the degraded signal which has been passed through the distorting system. In addition, the sampling rate of these speech files is needed in this model, which is either 8k or 16k, and the test signal should be speech-like (Malden Electronics Ltd, 2007).

There are several stages in the operations performed by PESQ, including Level

alignment, Input filtering, Time alignment, Auditory transform, Equalisation and Disturbance processing. The processing carried out by PESQ is illustrated in Figure 4.17.

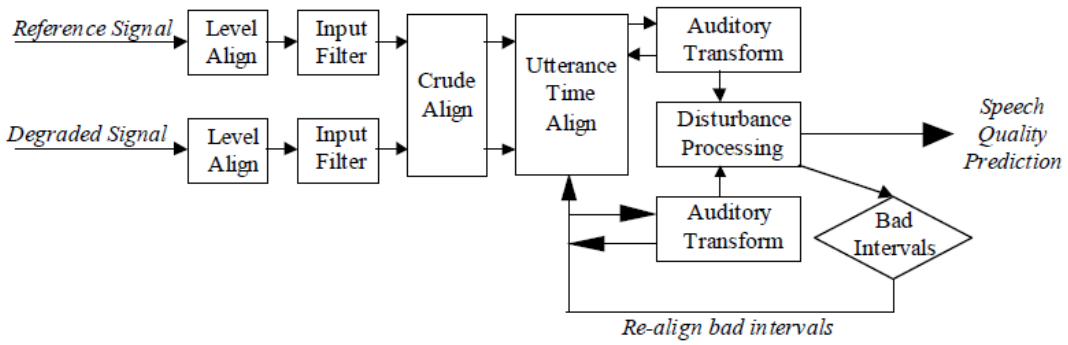

Figure 4. 17 Processing performed in PESQ

PESQ score is calculated according to ITU-T P.862, and PESQ P.862.1 gives a quality score on a MOS-like scale for narrowband listening. The aim of the amended recommendation ITU-T P.862.1 is to provide a single mapping from the raw P.862 score to the Listening Quality Objective Mean Opinion Score (LQO-MOS). The mapping from PESQ score to PESQ P.862.1 is computed as follows:

$$PESQP.862.1 = 0.999 + \frac{4.999 - 0.999}{1 + e^{-1.4945 \times PESQScore + 4.6607}} \qquad (4.13)$$

4.6 Summary

In this chapter Cachin's definition of steganographic security is discussed to identify the limitations of its usage in streaming media steganography. A new information-theoretic model of secure covert VoIP communication was proposed and proved to be a ε- secure stegosystem according to the information theory.

Covert VoIP communication algorithms were devised in this project to complementAES-128 encryption with data embedding and data extraction, which were examined to be applicable to covert VoIP communications.

Based on the new theoretical model, a covert VoIP communication system called StegPhone was developed in C++ programming code. The experimental set-up for covert VoIP communications is described in detail, and evaluation criteria, e.g. the perceptual evaluation of speech quality value and the signal-to-noise ratio, are also discussed. The performance measurements were carried out by using Digital Speech Level Analyser.

CHAPTER 5 True Random Key Generation and Chaotic Interval Selection for Covert VoIP Communications

This chapter explores the potential of hardware-based true random key generation and chaotic logistic map for innovative applications in covert VoIP communications over streaming media. It focuses on the investigation into how the read time stamp counter of the CPU as an entropy source can be used to generate true random numbers as secret keys for streaming media steganography underpinning covert VoIP communications. It then covers devising an interval selection algorithm to choose randomly data embedding locations in VoIP streams using random sequences generated from a logistic chaotic map. The initial parameters of the chaotic map and the selection of where to embed the secret message are negotiated between two communicating parties. It finally carried out experiments and security analysis using the Mann-Whitney-Wilcoxon (M-W-W) test, instead of parameter-based statistical test, to prove the security of the proposed steganographic algorithm for covert VoIP communications.

5.1 Introduction

Voice over IP (VoIP) communication systems have been embedded into an increasing number of industrial applications such as smart transportation systems and intelligent healthcare systems. When implementing VoIP applications in building systems or

developing innovative smart products to assist people, security issues need to be addressed urgently due to ever evolving cyber threats in recent years.

VoIP can be achieved on any networks based on an internet protocol (IP), such as the Internet, Intranet, local area networks (LANs) and wireless networks. VoIP applications include Personal Computer (PC) to PC connection, PC to Public Switched Telephone Network (PSTN) or PSTN to PC connection and PSTN to PSTN connection. The main services include voice services and real-time fax services over IP-based networks, interactive voice response (IVR) services implemented on the Web, and a variety of communication services such as E-mail and real-time telephone. VoIP services operate on an Internet protocol to transmit compressed voice samples as frames and messages as a group of bytes over an IP data network. In VoIP applications, voice from end-user equipment is converted into a signal level, digitised, compressed as voice payload and sent as IP packets.

VoIP transmits voice information over an IP network to realise real-time voice communication. The basic transmission process of VoIP includes collecting the original sender's voice, converting the original voice signal into a digital signal by analogue-to-digital conversion, compressing and encoding the digital signal through a voice compression algorithm, encapsulating the compressed voice data according to the standard of TCP/IP, and sending the encapsulated IP packets to the receiver over an IP network. The receiver decodes and decompresses the received voice data packets to obtain the original analogue voice signal, so as to realise the real-time communication of voice information on the network.

VoIP media streams are the dynamic flow of voice data packets which can be used as cover objects to build real-time steganographic systems with embedded VoIP for industrial applications such as new healthcare products to assist our aging population (Fridrich, 2014). Cryptography and steganography are expected to complement each other to improve the security of steganographic systems. As a steganographic message is embedded in VoIP media streams after encryption, a strong key is essential to ensure that the message it protects remains absolutely secure. However, the key used for encryption and decryption of the message is normally a pseudorandom number, which is not secure enough because the key is subject to compromise. Given enough time and computational powers, the key would be cracked by attackers: if multiple PCs work in parallel, the time is drastically shortened, and today's supercomputers should be able to find a pseudorandom key in about an hour (Stallings, 2017). A true random number based on hardware is a perfect seed for a strong key which can guarantee the security of steganographic systems.

A number of research have been conducted on the basic techniques of real-time steganographic systems with embedded VoIP, but the security of keys used for the systems has not received the attention it deserves. If the keys are broken, the systems become insecure, an eavesdropper can distinguish between the ordinary objects and the stego objects that contain the secret message, which means the secret message could be extracted from the stego objects. It would compromise the steganographic systems.

Efforts have been carried out to investigate the use of VoIP media streams as cover objects to build real-time steganographic systems with embedded VoIP for secure communication.

Aoki developed a semi-lossless steganographic VoIP technique for G.711 telephony speech (Aoki, 2010), with bandwidth improved from 24 bit/s to 400 bit/s, depending on the background noise signal level. Huang et al. proposed a high capacity steganographic algorithm for embedding data in various speech parameters of the inactive frames of low bit rate VoIP streams encoded with G.723.1 source codec (Huang, Tang, Yuan, 2011). Tian et al. suggested a method to improve the performance of VoIP steganography by adding some similarity between the hidden message and the cover object (Tian et al., 2012) to achieve a balance between steganography transparency and bandwidth, but the similarity limited the choice of hidden messages. Tian et al. improved the security of quantization-index-modulation steganography in low bit-rate VoIP speech streams (Tian et al., 2014).

Qi et al. proposed using discrete spring transform (DST) to eliminate the perceived redundancy in VoIP multimedia signals and improve speech quality (Qi et al., 2016). Liu et al. reported the use of a matrix embedding method to achieve VoIP steganography in linear predictive coding for low bit-rate speech codec (Liu et al., 2016). Janicki investigated pitch-based VoIP steganography using Speex voice codec (Janicki, 2016) to complement Aoki's work (Aoki, 2010). Tian et al. (2017) put forward a bitrate modulation steganographic algorithm with Hamming matrix VoIP encoding, but its practicality needed further study. Xin et al. suggested an adaptive audio steganographic algorithm for covert wireless communication, which was based on variable low bit coding (Xin et al., 2018).

More recently, Yi et al. studied separable and reversible data hiding in encrypted images using parametric binary tree labelling, achieving an average embedding rate up to 2.003 bpp (Yi et al., 2019). Zhou et al. reported a distortion design for secure

adaptive 3D mesh steganography, which relied on some effective steganalytic features such as variation of vertex normal (Zhou et al., 2019). Overall, these studies mainly focused on algorithm design for steganographic VoIP systems.

Until recently, only few references in the literature mentioned the keys for VoIP steganographic systems. This was the motivation behind this study.

5.2 Random Number Key Generation

Random number keys play a fundamental role in using cryptography and steganography to develop a real-time steganographic system with embedded VoIP. Two types of random number generators can be classified, namely true random number generators (TRNGs) and pseudorandom number generators (PRNGs). TRNGs produce random bits from random physical phenomena or noise sources. Such non-deterministic generators have limited efficiency in number generation rates due to restricted mechanisms for extracting bits from physical procedures. On the contrary, pseudorandom number generators are initiated by a relatively short key (seed) and their output is expanded into a long sequence of random bits using computational deterministic algorithms.

Compared with pseudorandom number generators，a hardware-implemented truly random key generator has limited efficiency in random number generation. It’s not suitable for generating a large scale of random numbers. As the random number in the covert VoIP communication system is used for key generation, a true random number generator has both enough capacity and better security, thus it is adopted in this research.

As for random number generation, Danger et al. developed a high speed true random number generator based on open loop structures in Field Programmable Gate Array (FPGA) (Danger et al., 2009), and the proposed architecture was generic as it was based on the open loop structure with no specific component such as Phase lock loop (PLL). Argyris et al. demonstrated the first compact real-time true random bit generator (TRBG) that exploited broadband chaotic signals emitted by a photonic integrated circuit (PIC) (Argyris et al., 2010). Bayon et al. presented a contactless and local active attack on ring oscillators (ROs) based TRNGs using electromagnetic fields (Bayon et al., 2012). Hirabayashi et al. built a model to realise true random one-time pad (OTP) encryption using DNA self-assembly (Hirabayashi et al., 2009). The first commercially available true random number generator that achieves bit production rates comparable with that of PRNGs is the Intel digital random number generator (DRNG) offered on new multicore chips in May 2012 (Stallings, 2017).

Several attempts have been made to study online evaluation of true random number generators. Veljkovic et al. conducted low-cost implementations of on-the-fly tests for random number generators in constant test sequence length (Veljkovic et al., 2012). Yang et al. performed a series of experiments to carry out the fly testing of true random number generators keeping track of a random walk (Yang et al., 2015). Their studies focus only on the implementation of the tests and no TRNG evaluation results are provided. Fischer et al suggested a method to assess the internal health of an oscillator based TRNG by measuring the jitter (Fischer et al., 2014). Hussain et al. reported the first online and hardware-based testing methodology that includes higher order randomness tests (other than the frequency test), and they claimed that the method is applicable to all physical unclonable functions families (weak and strong)

(Hussain et al., 2016). However, there is no hardware-based true random number generator available for real-time steganographic systems with embedded VoIP.

In summary, the security of keys used for VoIP steganographic systems has not been given much attention. To address the security issue, the remaining sections explore the potential of using hardware-based true random keys and advanced cryptography to improve the security of the real-time steganographic system with embedded VoIP.

5.3 Proposed Real-time Steganographic VoIP System

A real-time VoIP system with a steganographic algorithm to protect the data embedded in VoIP streams is devised in this research to achieve secure VoIP data communication. The system consists of four main modules: VoIP Communication, Random Key Generation, Selection of Embedding Locations, and Data Embedding and Data Extraction.

5.3.1 VoIP Communication

In this VoIP communication system, end-user terminals are connected to a VoIP proxy server on an IP local network, as shown in Fig. 5.1 A connection is established between two communicating parties using the SIP signalling protocol, followed by real-time audio communications with the media protocol, and the IP protocol handles the VoIP voice transmission over the IP network. All these protocols are implemented on the VoIP server, which is connected to end-user terminals.

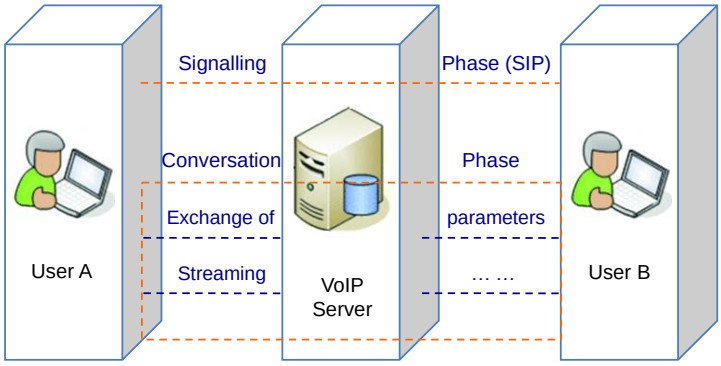

Figure 5. 1 VoIP communication system

The sender saves the secret message to be hidden into a .txt file, and then choose the file in the system to embed the secret message into the cover object. After extracting the received audio packets, the receiver obtains a .txt file which contains the original secret message.

The VoIP system is developed using C++ and MFC. Audio signal acquisition and playback is implemented by means of winmm.lib which is a multimedia API, and real-time transmission of streaming audio packets is based on jrtplib 3.9.1 library. The system uses the User Datagram Protocol (UDP) rather than Transmission Control Protocol (TCP) to establish low-latency and loss-tolerating connections between VoIP communicating parties.

The end-user's IP address is needed to initialise secure VoIP communication. Several parameters are then set for encoding, sampling and quantizing VoIP streams. In the communication system, VoIP audio samples are coded with single-channel PCM having a sampling rate of 8000 Hz, which are used as cover objects for steganography. Each speech sample is represented with 16 bits.

5.3.2 Random Key Generation

Cryptography and steganography can complement each other to improve the security of real-time steganographic systems with embedded VoIP by means of a set of keys. A secure steganographic system requires strong random keys. The random nature of the chosen number means it could lie anywhere on a virtually endless number line. A poor random number generator generates pseudorandom numbers, *i.e.* numbers generated by a predictable process. Most of the keys are generated by software that uses software-based random number generators. Keys based on pseudorandom numbers are subject to compromise, meaning that the data encrypted with such keys is not secure.

The one time pad – the only provably secure encryption system – uses as much key material as ciphertext and requires that the keystream be generated from a truly random process (Jun et al., 1999). Generating a strong key ultimately relies on a good source of entropy. Today's operating systems provide non-physical true random number generators which are based on hardware events. Hardware entropy sources are preferable because they yield one bit of entropy for every bit of seed.

In this research, an entropy source-based method that uses the read time stamp counter of the CPU is devised to generate true random numbers as dynamic keys for the real-time steganographic system with embedded VoIP.

Figure 5.2 illustrates the true random number generator in which the read time stamp counter of the CPU is obtained to represent the clock cycles since the CPU starts. The instruction returns in registers EDX: EAX the count of ticks from the processor reset. A random number is generated by first obtaining the real-time system compiler

(RTSC) of the CPU and using a linear feedback shift register. The RTSC aims to provide an operating system aware compiler that allows for a generic manipulation of the real-time system architecture of a given real-time application. The RTSC is added to fill the 64-bits. One of the randomly large integers can be chosen to generate a large prime number. The prime number is then used to generate keys. Since the seed is a true random number, the key created with it is also true random.

```
__int64 CDiffieHellman::GetRTSC( void )
{
        int tmp1 = 0;
        int tmp2 = 0;
        __asm
        {
                RDTSC;
                mov tmp1, eax;
                mov tmp2, edx;
        }
        return ((__int64)tmp1 * (__int64)tmp2);
}
unsigned __int64
CDiffieHellman::GenerateRandomNumber(void)
{
                        static unsigned long rnd = 0x41594c49;
        static unsigned long x      = 0x94c49514;
        LFSR(x);
        rnd^=GetRTSC()^x;
        ROT(rnd,7);
        return (unsigned __int64)GetRTSC() + rnd;
}
```

Figure 5. 2 Partial pseudo-code of true random number generation.

The following prediction of optimum guessing attack cost proves the security of hardware entropy source-based true random keys used for the real-time steganographic system with embedded VoIP.

Assuming that an attacker wants to guess a secret number among a set of *n* random numbers with the probability distribution

$$P = \{p_1, p_2, \ldots, p_n\} \tag{5.1}$$

and guesses a maximum of *i* possibilities for a given secret number in order to minimise the expected number of guesses per successful recovery, the optimum value of *i* can be anywhere in the range $1 \le i \le n$, depending on the probability distribution *P*.

If the secret number is not any of the *i* – 1 most likely numbers, the probability of making *i* guesses is $1 - \sum_{j=1}^{i-1} p_j$ for $1 \le j \le i - 1$. The expected number of guesses for the attack can be written as (Stallings et al., 2019)

$$NoG = p_1 + 2p_2 + \ldots + (i-1)p_{i-1} + i\left(1 - \sum_{j=1}^{i-1} p_j\right) \qquad (5.2)$$

The probability that the attack is successful is $\sum_{j=1}^{i} p_j$ if and only if the secret number is one of the *i* most likely possibilities. Thus, the expected guess per success is given by

$$G_i(P) = NoG \Big/ \sum_{j=1}^{i} p_j \qquad (5.3)$$

Given that the hardware-entropy of *P* corresponds to the exact expected guess (measured in bits) needed to perform the optimum guessing attack or over-estimates this guess by at most one bit, it has

$$G_i(P) \geq G_i(P'). \qquad (5.4)$$

For a probability distribution on a set of *n* possibilities $P' = \{p_1, p_1, \ldots, p_1, 1 - tp_1, 0, \ldots, 0\}$, p_1 occurs $t = \frac{1}{|p_1|}$ times. According to (5.3), for $1 \le i \le t + 1$, the numerator of $G_i(P')$ is given by

$$\sum_{k=0}^{i-1}\left(1 - kp_1\right) = i - \frac{i(i-1)}{2} p_1 = ip_1\left(\frac{1}{p_1} - \frac{i-1}{2}\right) \qquad (5.5)$$

For $1 \le i \le t$, the denominator of $G_i(P')$ is ip_1. So

$$G_i(P') = \frac{ip_1\left(\frac{1}{p_1} - \frac{i-1}{2}\right)}{ip_1} = \frac{1}{p_1} - \frac{i-1}{2} \qquad (5.6)$$
$$\ge \frac{1}{p_1} - \frac{1}{2}\left(\frac{1}{|p_1|} - 1\right) \ge \frac{1}{2p_1} + \frac{1}{2}$$

Based on (5.4), for $1 \le i \le n$, it has

$$G_i(P) \ge G_i(P') \ge \frac{1}{2p_1} + \frac{1}{2} \qquad (5.7)$$

For hardware-entropy random numbers, *P* is a uniform distribution, and the expected guess per success achieves this sharp lower bound. That means that the optimum guessing attack cost is extremely high for hardware entropy source-based true random keys.

5.3.3 Selection of Embedding Locations

Chaos is a stochastic phenomenon of nonlinear deterministic system in the nature. The uncertainty of random sequence is resulted from the internal factor of a chaotic dynamical system. A chaotic map is extremely sensitive to the initial conditions and the parameters of the chaotic map. So a mass of noise-like but determinate random sequences can be obtained from a chaotic map. The random sequence can be reproduced from the same chaotic map with the constant initial condition. In comparison with other pseudo random sequence generation algorithms such as Mersenne Twister method, using a logistic map to generate a random sequence is straightforward and convenient to implement.

In the proposed scheme, a series of random sequences generated from a logistic chaotic map are used to choose data embedding locations in VoIP streams. The utilization of chaotic map makes data embedding in VoIP streams randomly, and it is unlikely to predict the initial conditions of random sequences. So the properties of the chaotic map can increase the security of covert communications. To meet the real-time requirement, it needs to minimize the time to generate random sequences. Meanwhile, it is necessary to know the initial parameters with infinite precision for sensitivity of the chaotic map, so that the initial parameters can be transmitted securely between the communicating parties in the conversation phase.

A logistic map is one of the most popular models for discrete nonlinear dynamical systems. The map is popularized in a seminal paper by the biologist Robert May, in part as a discrete-time demographic model analogous to the logistic equation first created by Pierre Francois Verhulst (Weisstein, 2003). A logistic map is given by

$$x_{n+1} = \mu x_n (1 - x_n) \qquad (5.8)$$

where x_0 is the initial ratio of the population to the maximum population at year 0, x_n denotes the value of x_0 after n iterations, a number between 0 and 1, and the ratio of existing population to the maximum possible population after n years, and μ is a positive number which stands for a combined rate for reproduction and starvation.

Figure 5.3 depicts a bifurcation diagram of a logistic map when $x_0 = 0.52$. As Fig. 5.3 shows, when $\mu \in (0,1]$, the value of x is equal or close to 0. When $\mu \in (1,3]$, x quickly approaches the value of μ-1/μ. It shows chaotic characteristics when μ varies in the range (3.57, 4]. When $\mu = 4$, x becomes increasingly chaotic. Figure 5.4 shows the ergodic property of chaos, when $x_0 = 0.52$ and $\mu = 4$ in the equation of a logistic map.

As can be seen from Fig. 5.4, the value of x_n randomly falls in the range (0, 1] as n increases between (0,10000].

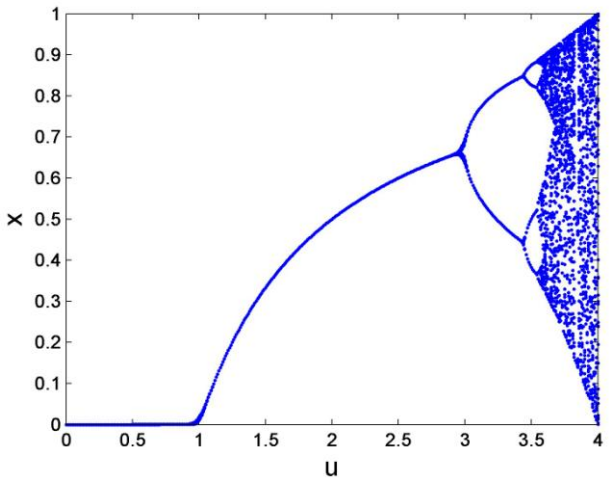

Figure 5. 3 Bifurcation diagram of a logistic map when $x_0 = 0.52$.

Tent map is also a discrete-time dynamical system model. The original formula for the Tent map can be written as:

$$\begin{cases} x_{n+1} = \mu x_n & 0 < x_n < \frac{1}{2} \\ x_{n+1} = \mu(1 - x_n) & \frac{1}{2} < x_n < 1 \end{cases} \tag{5.9}$$

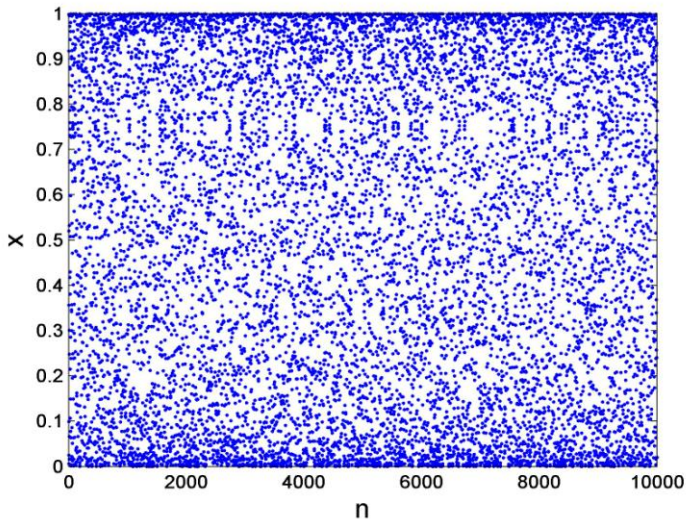

Figure 5. 4 Values of x_n with n increasing when $x_0 = 0.52$ and $\mu = 4$.

To extend the mapping range, the improved Tent map is obtained:

$$\begin{cases} x_{n+1} = \mu x_n & 0 < x_n < \dfrac{a}{2} \\ x_{n+1} = \mu(a - x_n) & \dfrac{a}{2} < x_n < a \end{cases} \tag{5.10}$$

In equation (5.10), $x \in (0,a)$, $m \in (0,2)$, and $a \in R$. For the Tent map, the parameters are a = 16, μ = 1.99, and x_0 = 0.552. The parameter values of the logistic map are x_0 = 0.52 and μ = 4.

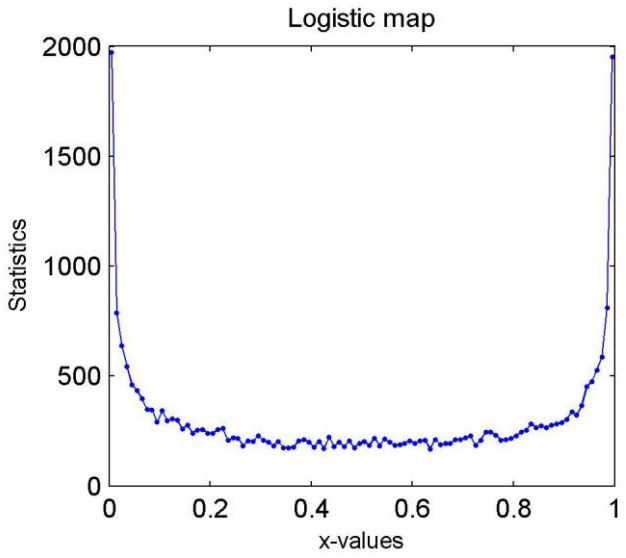

Figure 5. 5 Statistical distribution of numbers (Y axis) generated from the logistic map (X axis: x_n).

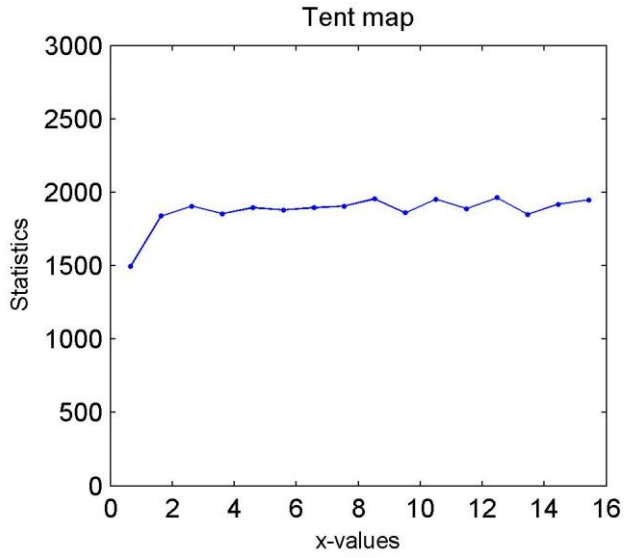

Figure 5. 6 Statistical distribution of numbers (Y axis) generated from the improved Tent map (X axis: x_n).

As Figs. 5.5 and 5.6 show, after 30000 iterations, the statistical distribution of the numbers generated from the logistic map is 'U' shape distributed. And the statistical distribution of the numbers from the improved Tent map is almost uniformly distributed.

Since x is between 0 and 1, some adjustments need to be made to obtain a random sequence of integers from the logistic map. Suppose x_0 and μ are given as certain values in Equation (5.8), a sequence of x can then be yielded and denoted by X as

$$X = \{x_i \mid i = 0,1,2,...,n\} \tag{5.11}$$

and $$R = \{r_i = x_i \times 1000 \pmod{p} + 1 \mid i = 0,1,2,...,n\} \tag{5.12}$$

where R is the embedding location set, r is the embedding location, and p is the largest interval. To get x = 16, the adjustment for the Tent map is x_n = floor(x_n) + 1.

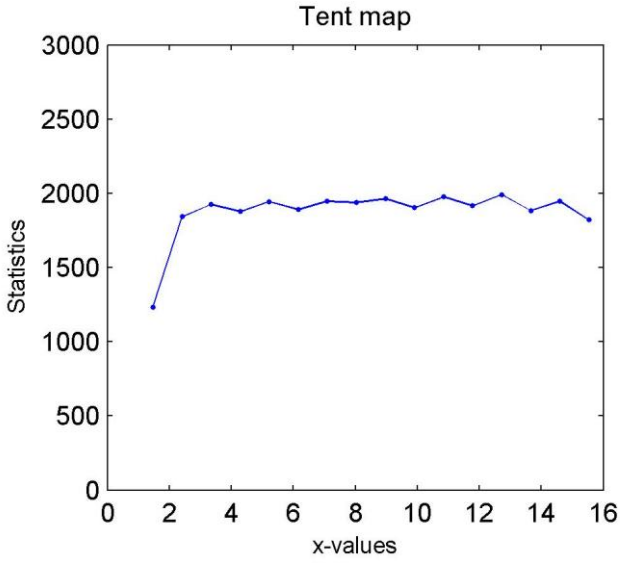

Figure 5. 7 Statistical distribution of numbers (Y axis) generated from the improved tent map after adjustment (X axis: x_n).

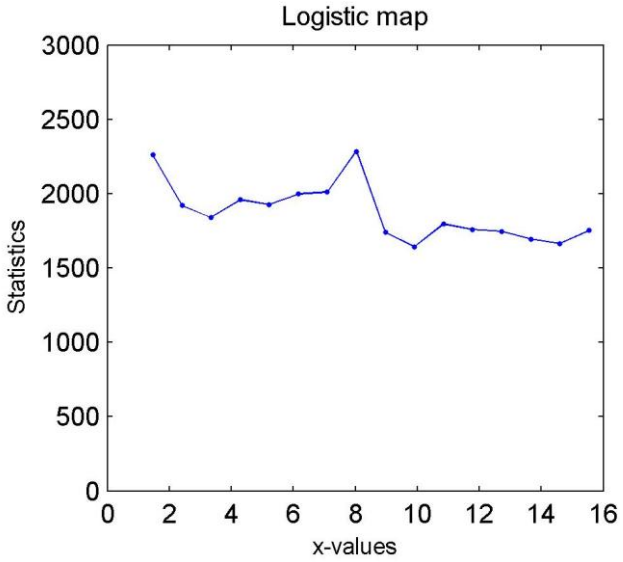

Figure 5. 8 Statistical distribution of numbers (Y axis) generated from the improved logistic map after adjustment (X axis: x_n).

As Fig. 5.7 shows, after adjustment the statistical distribution of numbers from the improved tent map is still uniformly distributed. As can been seen from Fig. 5.8, when x is in (Fridrich, 2014; Aoki, 2008) the numbers generated are around 2000, and the numbers from the logistic map are almost evenly distributed too.

The sequence of R is utilized in the proposed covert VoIP communications. p is an integer, which represents the largest interval, and the value of p is in the range of 2 to 35 (Tang et al., 2014). The PESQ scores and SNR values are stable before the embedding interval reaches 35. r_i is used to determine the data embedding locations in VoIP streams to embed the bitstream of secret data.

5.3.4 Data Embedding

A new steganographic algorithm is devised for the real-time VoIP communication system developed in this study. In the system, AES-128 is used to encrypt the secret message to be hidden before being embedded into VoIP streams to improve secrecy.

Since VoIP streams are transmitted over the public network, the impact of delay and packet loss is inevitable. Given that the extraction and decryption of the secret message can be affected by packet loss, the encrypted message is segmented into distinct blocks before data embedding, so that the original secret message is separately extracted and decrypted from independent VoIP packets.

Due to the real-time requirements for VoIP communication, it is necessary to shorten the time required to encrypt the secret message. AES-128 encryption is a block cipher which takes the plaintext block size of 128 bits (or 16 bytes), and the encryption time is less than 0.1ms. In order to embed the secret message as much as possible in a VoIP packet and reduce the encryption time to meet the needs of real-time communication, the length of the secret message embedded in each single VoIP packet should be a multiple of 16 bytes and it must be smaller than the embedding capacity of a single audio packet.

The encrypted secret message is segmented and then embedded in VoIP streaming packets at various embedding capacities and different embedding locations. In the VoIP system, the size of the VoIP packet is set to be 4096 bytes. The first 16 bytes of the first VoIP packet is used to embed the length of the secret message to be hidden (LoM), and the remaining of the first packet and the other VoIP packets are used to embed the secret message itself. The size of the remaining of the first VoIP packet is 4080 bytes, and the data embedding capacity is up to 510 bytes (12.5%). To reduce the encryption time, the first 496 bytes of the secret message are embedded in the first VoIP packet as it is a multiple of 16 bytes. As for the other VoIP packets, 512 bytes of the secret message are embedded in each VoIP packet. The remaining of the secret message is embedded in the last VoIP packet with the size of LLoM. The

LLoM value may not be a multiple of 16 bytes, so it is possibly necessary to adjust it to a multiple of 16 bytes in some cases (Peng et al., 2020).

In order to achieve variable embedding capacity, a parameter R is set as the data embedding interval to minimise the impact of data embedding on voice quality in different environments and at varying lengths of the secret message. The larger the embedding interval is, the less the secret message could be embedded into each VoIP packet. For simplicity, 16 bytes of VoIP streams have a data embedding capacity of two bytes. When the value of R is set to be one, the interval at which the secret message is embedded is two bytes, i.e. one byte of the secret message is randomly embedded in 16 bytes of VoIP streams. If the interval is set to two, then the secret message is randomly embedded in VoIP streams at an interval of three bytes; in this case 22 bytes of VoIP streams contain one byte of the secret message, and so on.

A new data embedding algorithm is used to embed secret data in VoIP cover-speech streams. It is based on bit substitution with payloads using a random sequence to determine the data embedding location, i.e. where to embed the bitstream of secret data. At the beginning, there are two choices: either embedding secret data into an active or inactive speech period of VoIP streams. If data embedding in an active speech period is chosen, the embedding algorithm waits until the active speech period starts. An important parameter *ActiveChoice* is set to represent the choice. If the value of *ActiveChoice* is ‘false’, secret data is embedded in inactive speech periods, and the algorithm continues with inactive speech periods. If the value of *ActiveChoice* is set to ‘true’, the bitstream of secret data is embedded in active speech periods.

Assuming user A wants to send L bits of secret data M to user B, the secret data is described as $M = \{m_i | \ i = 0,1,2,...,L\text{-}1\}$. If the number of samples in the piece of speech of each packet is N, the least significant bit set of samples can be denoted by $S = \{s_i | \ i = 0,1,2,..., N\text{-}1\}$.

The steganographic algorithm devised for the real-time VoIP system is depicted as follows:

Step A: Obtain a random x_i from the logistic chaotic map described in the previous section, calculate $R_i = x_i \times 1000 \pmod{p} + 1$ as the data embedding interval to decide the data embedding location in VoIP streams.

Step B: first embed the length of the secret message (LoM) in VoIP media streams, and let length = LoM.

Step C: compute the length of the secret message hidden in the first packet (m_1) and the length of the secret message hidden in other packets (m_k).

If LoM < m_1,

encrypt $M(0, m_0 - 1)$ to form $E(0, m_0 - 1)$, and embed $E(0, m_0 - 1)$ in the bit stream $BS = \{b(0), b(1), \ldots, b((m_0 - 1) * 8)\}$.

If $b(i) = 0$, $V(k) = V(k)\&0xfe$.

If $b(i) = 1$, $V(k) = V(k)|0x01$.

k = k + R,

length = 0,

end.

If LoM > m_1,

encrypt $M(0, m_1 - 1)$ to form $E(0, m_1 - 1)$, and embed $E(0, m_1 - 1)$ in the bit stream $BS = \{b(0), b(1), \ldots, b((m_1 - 1) * 8)\}$.

If $b(i) = 0$, $V(k) = V(k)\&0xfe$.

If $b(i) = 1$, $V(k) = V(k)|0x01$.

k = k + *R*,

length = length − m_1.

where m_0 is the whole size of the secret message to be embedded, m_1 denotes the length of the secret message embedded into the first packet, m_k represents the length of the secret message embedded into other packets, m_n denotes the length of the secret message in the last packet. & is a bitwise AND operation, | is a bitwise OR operation, and 0x is hexadecimal.

Step D:

If length > m_k,

encrypt $M(m_1, m_1 + m_k - 1)$ to form $E(m_1, m_1 + m_k - 1)$, and embed $E(m_1, m_1 + m_k - 1)$ in the bit stream $BS = \{b(0), b(1), \ldots, b((m_k - 1) * 8)\}$.

If $b(i) = 0$, $V(k) = V(k)\&0xfe$.

If $b(i) = 1$, $V(k) = V(k)|0x01$.

k = k + *R*,

length = length − m_k,

repeat Step C until length < m_k.

Step E: compute the length of the secret message hidden in the last packet (m_n).

Encrypt *M*(LoM− length, LoM− 1) to form *E*(LoM− length, LoM− length + m_n − 1), and embed *E*(LoM−length, LoM−length+ m_n −1) in the bit stream *BS* = {*b*(0), *b*(1), . . . , *b*((m_n − 1) * 8)}.

If *b*(i) = 0, *V* (k) = *V* (k)&0xfe.

If *b*(i) = 1, *V* (k) = *V* (k)|0x01.

k = k + *R*,

length = 0,

end.

5.3.5 Data Extraction

The extraction of the secret message, steganographically embedded in VoIP streams using the data embedding algorithm above, from the stego VoIP streams received on the receiver side is the inverse process of the data embedding algorithm. The corresponding extraction algorithm is used to retrieve the secret message encrypted with AES, and decrypt it with the same secret keys to obtain the original secret message from the stego VoIP packets.

In the VoIP conversation phase (Fig. 5.1), the receiver obtains the initial parameters of a logistic map, which is used to generate random sequences that randomly choose data embedding locations, and the value of *ActiveChoice* which determines where to extract the secret data hidden in VoIP streams. The same initial parameters and the value of *ActiveChoice* enable the receiver to successfully retrieve the secret data. The

extraction process is a reverse phase of the data embedding process. After receiving an audio packet, the least significant bit set of samples can be denoted by $S' = \{s'_i \mid i = 0, 1, 2, ..., N-1\}$. The logistic chaotic map generates a corresponding random number x_i that decides the extraction location.

The following steps are performed to extract the original secret data on the receiver side.

Step 1: The initial value of j is 0, and s'_0 is the least significant bit of the first sample in the chosen piece of speech.

Step 2: Generate a random number x_i from a logistic chaotic map, and calculate r_i to decide the extraction location.

Step 3: Suppose the least significant bit of the current sample is s_j', if $j+r_i < N$, get s_{j+ri}' as m_i, then perform $j = j+r_i$, $i = i+1$, repeat Step 2 and Step 3 until the end of the current piece of speech S', i.e., $j+r_i >= N$.

Step 4: Play audio, receive the next audio packet, and repeat Steps 1-3 until the completion of extracting the secret message M.

5.4 Experiments

5.4.1 Measurements of Interest

To assess a real-time steganographic system with embedded VoIP, there are measurements of interest as follows:

A. Embedding capacity is defined as the maximal number of bits that can be embedded in the cover object (e.g. VoIP streams) using a given steganographic algorithm.

B. Embedding efficiency is defined as the number of embedded bits per unit distortion: <message size> / <changes to cover>.

C. Steganographic capacity is defined as the maximal length of a steganographic message that can be undetectably embedded in a given cover object (Cox et al., 2008).

D. Maximum capacity is defined as the maximal capacity achievable before risking detection.

E. VoIP speech and signal quality metrics include perceptual evaluation of speech quality (PESQ) and signal-to-noise ratio (SNR) (Huang et al., 2016).

5.4.2 Experimental Set-up

VoIP communication experiments were designed to assess the security and effectiveness of the real-time steganographic system with embedded VoIP using the proposed steganographic algorithm. Performance measurements were carried out by means of the state-of-the-art network equipment Digital Speech Level Analyser (DSLA).

In the experiments, VoIP speech samples in PCM format encoded with G.711 codec were employed as cover objects for real-time VoIP communications. The secret message to be hidden was encrypted with the keys generated from the true random

number generator detailed in the previous section, and was then segmented into parts which were embedded into VoIP packets. Performance was evaluated by comparing the original VoIP streams with the stego VoIP streams to assess the imperceptibility of the resulting stego VoIP streams. The data embedding capacity was calculated for each set of experimental results. DSLA was used to measure the SNR value and the PESQ score of the speech samples, which are two important parameters for performance evaluation. The figure in the previous chapter shows a diagram of performance measurements using DSLA for the real-time steganographic system with embedded VoIP.

The DSLA and its user interface have been designed to provide access to the SNR value and PESQ score, either directly from the analogue connection or from recorded speech files. It performs ITU-T P.862 objective speech quality scoring plus improved Mean Opinion Score prediction according to ITU-T P.862.1. This is a fully conformant implementation of PESQ as defined in ITU-T P.862, and can be used to measure the performance of the proposed steganographic algorithm.

5.4.3 Signal Quality

The signal-to-noise ratio is a measure which compares the level of a desired signal to a background noise, and it is one of the most commonly used measurements of speech quality in the field of VoIP communication. In the experiments, larger SNR values mean better quality of the stego audio streams preserving the original audio streams. The SNR value can be calculated using the equations below:

$$SNR = 20\log(\frac{255}{\sqrt{MSE}}) \qquad (5.13)$$

$$MSE = \frac{1}{N \times M} \sum_{i=0}^{N-1} \sum_{j=1}^{M-1} \left(\begin{matrix} PV\ of\ \text{cover}(i,j) \\ -PV\ of\ \text{stego}(i,j) \end{matrix} \right)^2 \qquad (5.14)$$

where PV is the pixel value, and the mean squared error (MSE) is a square value of the difference between the pixel value of cover audio streams and the pixel value of stego VoIP streams. Through the aforementioned formulae, it can be seen that the SNR value is inversely proportional to the MSE value. If MSE is equal to zero, SNR becomes infinite sequentially, meaning that no distortion occurs after data embedding.

5.4.4 Speech Quality

A modern communication system cannot be reliably assessed by conventional engineering metrics such as SNR alone. One solution to measuring customers' perception of the quality of the communication systems is to conduct subjective tests involving panels of human subjects. However, these types of tests are expensive and unsuitable for real-time monitoring applications. Perceptual evaluation of speech quality provides an objective measurement which predicts the results of subjective listening tests on audio communication systems. To measure speech quality, PESQ uses a sensory model to compare the original signal with the degraded signal of the communication system.

The result of comparing the reference signal and the degraded signals is a quality score. This score is analogous to the subjective 'Mean Opinion Score' (MOS) measured using panel tests according to ITU-T P.800. PESQ takes into account several sources of signal degradation, including coding distortions, packet loss,

transmission errors, delay and variable delay and the filtering in analogue network components. PESQ has not considered the subjective effect of level changes in the network, echo and the effect of round-trip delay on conversation (DSLA, 2013).

PESQ score is calculated according to ITU-T P.862. PESQ P.862.1 gives a quality score on a MOS-like scale for narrowband listening (ITU-T Recommendation, 2001). The purpose of the amended recommendation ITU-T P.862.1 is to provide a single mapping from the raw P.862 score to the Listening Quality Objective Mean Opinion Score (LQO-MOS). The mapping from PESQ score to PESQ P.862.1 is computed as follows:

$$PESQ\,\mathrm{P.862.1} = 0.999 + \frac{4.999 - 0.999}{1 + e^{-1.4945 \times PESQScore + 4.6607}} \qquad (5.15)$$

5.5 Results and Discussion

Each real-time VoIP experiment was repeated 12 times to obtain the average measurement results for each VoIP data streams sample, so as to assess the effectiveness and security of the new steganographic algorithm.

For comparison purposes, two sets of VoIP communication tests were implemented with and without data embedding. Performance and security comparisons were made between the original VoIP streams and the stego VoIP streams as a result of embedding the encrypted secret message in VoIP streams using the new steganographic algorithm (with hardware-based true random keys). Real-time VoIP communications tests were also conducted using the proposed algorithm at variable data embedding intervals. VoIP communications with the algorithm were examined

by varying the hidden message size. The experimental results are discussed in detail below.

5.5.1 Spectrums of VoIP Streams

Figure 5.9 shows the waveforms of the original VoIP streams and the stego VoIP streams (containing steganographically embedded data) in the time domain, respectively. As can be seen, there was little distortion between the original VoIP streams (top figure) and the stego VoIP streams (bottom figure); distortion was probably caused by background noise. Listening tests indicated that the human ear could not distinguish any difference between these two types of VoIP streams. These results suggest that data embedding has no or little impact on the real-time steganographic system with embedded VoIP detailed in Section 5.3.

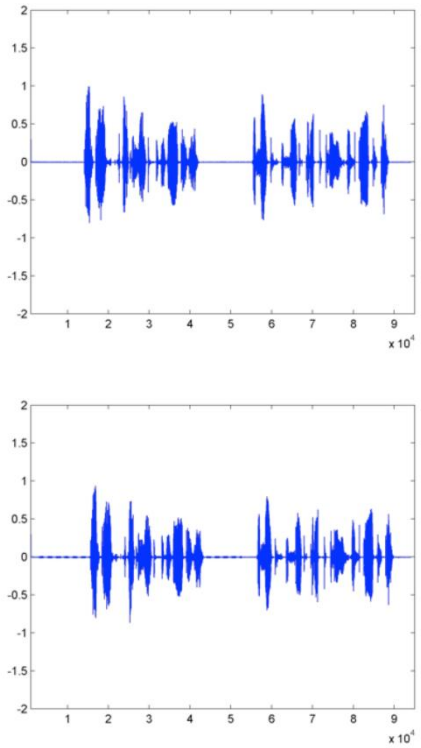

Figure 5. 9 Waveforms of the original and stego VoIP streams.

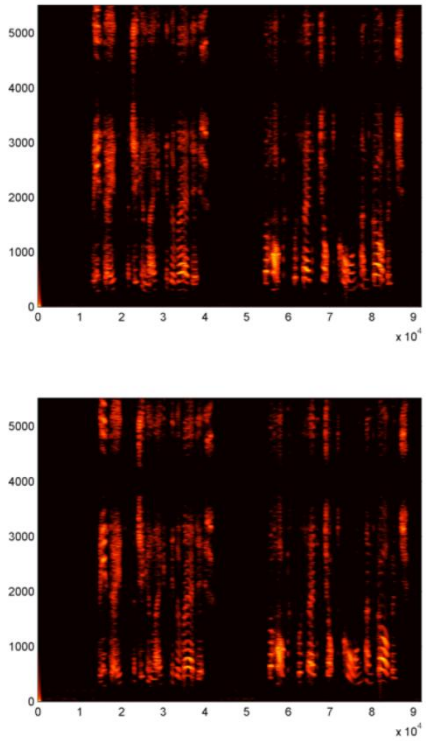

Figure 5. 10 Spectrograms of the original and stego VoIP streams

Figure 5.10 shows the spectrograms of the original VoIP streams (top) and the stego VoIP streams (bottom) in the frequency domain, respectively. The spectrogram is the result of calculating the frequency spectrum of sound. As the figure shows, there were slight differences in these spectrograms between the original and stego VoIP streams. It means that the proposed steganographic algorithm had little effect on the spectrogram of the real-time steganographic system with embedded VoIP. This finding is in good agreement with the waveform results as shown in Fig. 5.9.

5.5.2 Performance Comparisons

For each testing, 12 repeated experiments were carried out to obtain the average PESQ and SNR values using the experimental method described in the previous section.

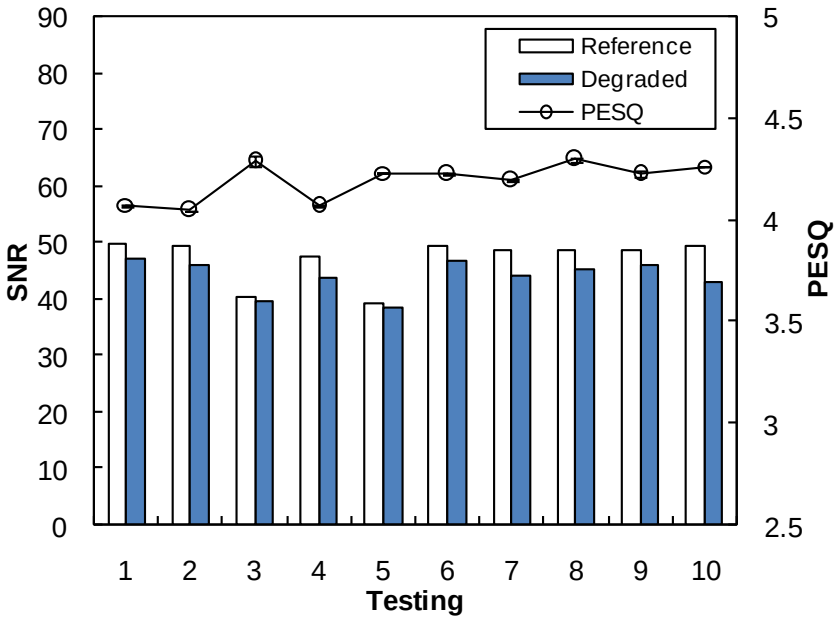

Figure 5. 11 Speech quality results of the original VoIP streams

Figure 5.11 shows the speech quality results of VoIP streams in the absence of data embedding (no secret message embedded in VoIP streams). The PESQ scores and SNR values of the VoIP streams samples were measured using DSLA. To evaluate the performance of VoIP communications, the streams samples recorded on the sender side were used as the reference (denoted as 'Reference'), and the VoIP streams samples on the receiver side were the degraded samples (denoted as 'Degraded') in the speech quality test.

The figure displays that the average PESQ scores of the original VoIP streams samples were between 4 and 4.5. It also shows comparisons in the average SNR between the reference and degraded samples, with the variance in SNR values being estimated to be 3.1 approximately. The results suggest that VoIP communications with high speech quality achieve in a real-time manner.

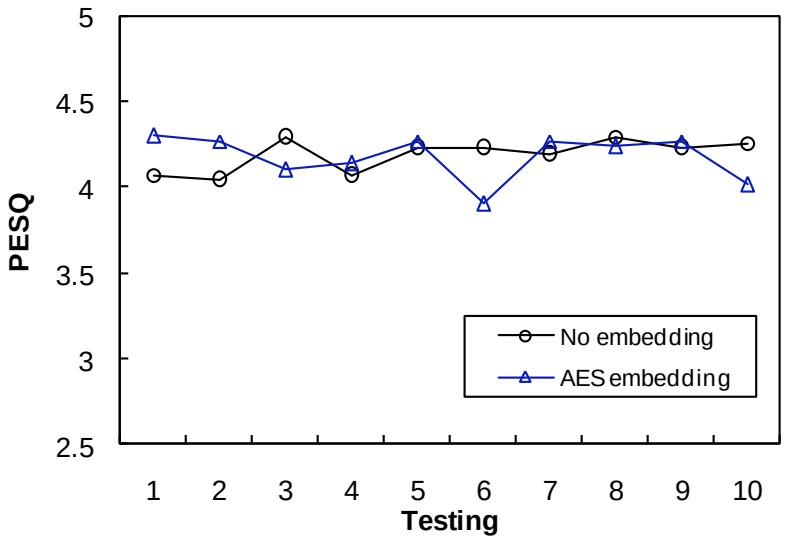

Figure 5. 12 Mean PESQ values for the original and stego VoIP streams.

Table 5. 1 PESQ values for the original and stego VoIP streams

Samples	Max	Min	Mean	Variance
Original VoIP streams	4.30	4.05	4.19	0.01
Stego VoIP streams (AES embedding)	4.31	3.90	4.18	0.02

Figure 5.12 and Table 5.1 shows the PESQ values for the original VoIP streams (denoted as 'No embedding') and the stego VoIP streams with the hidden message encrypted with AES before embedding using the proposed steganographic algorithm ('AES embedding'). The tests were run on the VoIP communications test bed as shown in the previous chapter, where the 15-second original streams samples were used as cover objects. Statistical results were obtained for steganographic communications experiments conducted at the same data embedding rate of 2 bits/frame. As Fig. 5.12 shows, the average PESQ mean values of the stego VoIP streams samples were between 4 and 4.5, close to the average PESQ value of the original VoIP streams samples, indicating an effective real-time steganographic system with embedded VoIP.

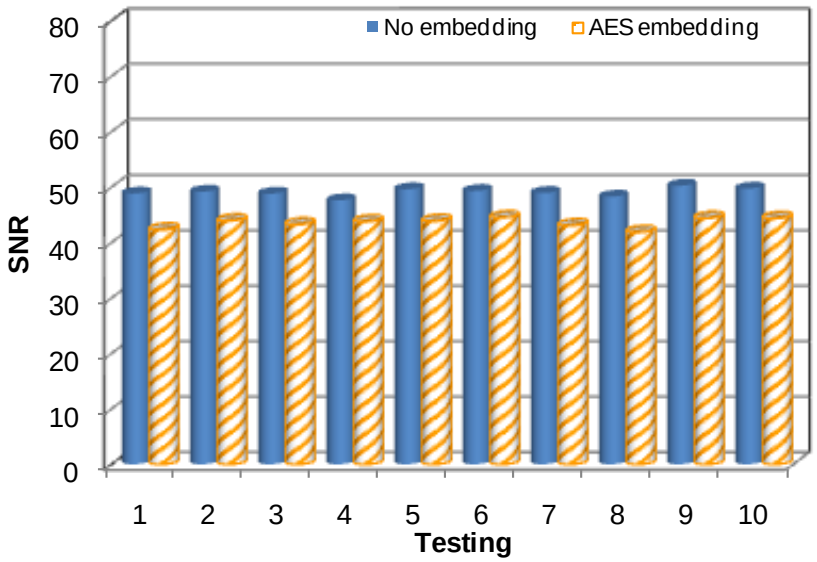

Figure 5. 13 Mean SNR values between the original and stego VoIP streams.

Table 5. 2 Mean SNR values between the original and stego VoIP streams.

Samples	Max	Min	Mean	Variance
Original VoIP streams	50.23	47.60	49.05	0.56
Stego VoIP streams (AES embedding)	44.47	41.83	43.50	0.81

Figure 5.13 and Table 5.2 shows a comparison of the mean SNR values between the original VoIP streams and the stego VoIP streams with the hidden message encrypted with AES before data embedding using the proposed steganographic algorithm. The average SNR value of the original VoIP streams was around 49.0, and that of the stego VoIP streams was about 43.5. This means the variance in SNR, the difference in the mean SNR value between the original and stego VoIP streams, is 5.5, close to that of the original VoIP streams (3.1), indicative of little degradation in speech quality caused by steganographically data embedding.

Figure 5.14 shows the 3D waveform in the time-domain and the spectrums in the frequency-domain of the original VoIP streams. Figure 5.15 shows the 3D waveform

of the stego VoIP streams, taken from real-time VoIP communications using the proposed steganographic algorithm. As Figs. 5.14 and 5.15 show, there were almost no differences in the 3D waveforms and spectrums. This implies that the new steganographic algorithm had no or little impact on the original VoIP streams in the time and frequency domains.

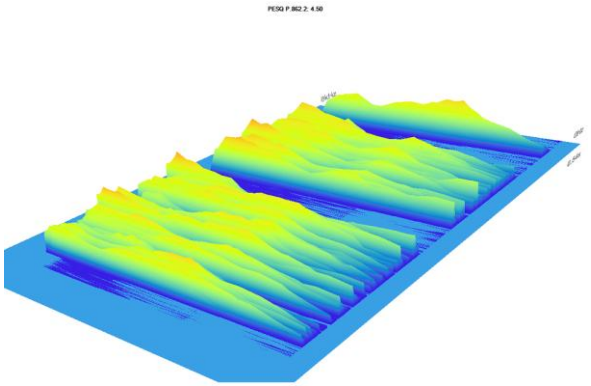

Figure 5. 14 3D waveform and spectrum of the original VoIP streams.

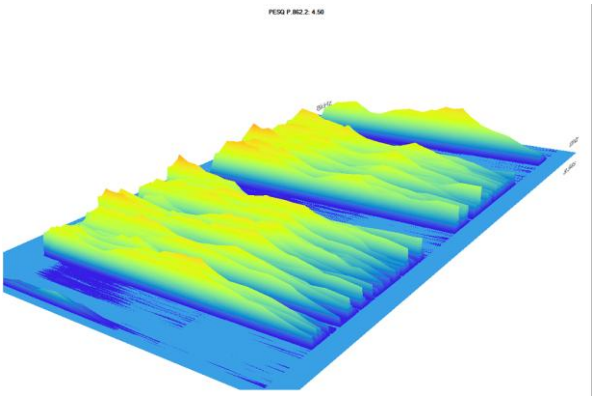

Figure 5. 15 3D waveform and spectrum of the stego VoIP streams

5.5.3 Statistical Undetectability Analysis

Steganographic communication is intended to conceal the existence of a steganographic message hidden in innocuous transmissions such as VoIP over the network. Statistical undetectability is normally used to evaluate the security of a

steganographic system (Fridrich, 2014), i.e. a measure of how difficult it is to reliably determine the existence of a secret message hidden in a cover object. In other words, a secure steganographic system means a statistically undetectable system.

Instead of normal statistical analysis, the Mann-Whitney-Wilcoxon (M-W-W) test was used in this study to perform statistical undetectability analysis to evaluate the security of the proposed steganographic algorithm. The test is one of the best-known non-parametric significance tests that assess whether two independent samples of observations come from the same distribution (Neter et al., 1993). A comparison of the probability distributions between the original and stego VoIP streams samples reveals whether the differences in the probability distributions are almost indistinguishable.

According to the principles of the M-W-W and statistics, when the sample sizes are sufficiently large (at least 12 each), the test is based on the standardized test statistic (Neter et al., 1993). In an examination of the proposed steganographic algorithm for the real-time steganographic system with embedded VoIP, the test statistic can be computed below:

First, combine the n_1 sample observations from Population 1 (*i.e.* the original VoIP streams samples) and the n_2 sample observations from Population 2 (*i.e.* the stego VoIP samples with the hidden message), and array the combined data in an ascending order;

Secondly, assign ranks to the combined observations (starting with 1 for the smallest observation);

Finally, sum the ranks for the n_2 sample observations from Population 2, and denote this sum by S_2.

The test statistic z^* is given by

$$z^* = \frac{S_2 - E\{S_2\}}{\sigma\{S_2\}} \qquad (5.16)$$

where $E\{S_2\}$ and $\sigma\{S_2\}$ are the mean and square root of variance of the sampling distribution S_2 that is the combination of the two samples of observations to be assessed.

To improve the precision of statistical undetectability analysis, the sizes of samples (data points) were 401 and 401 in computation of the test. Table 5.3 shows the test results and the parameters used for comparing the probability distribution drawn from the original VoIP streams samples to that drawn from the stego VoIP streams samples that steganographically embed the secret message using the new steganographic algorithm.

Table 5. 3 Undetectability analysis results using M-W-W test

Test	Sample size, n_1	Sample size, n_2	Sum, S_2	Mean of variance, $E\{S_2\}$	Variance, $\sigma^2\{S_2\}$	Test statistic, z^*	H
Test 1	401	401	160054	161001.5	10760267	-0.2888	H_0
Test 2	401	401	159060	161001.5	10760267	-0.5919	H_0
Test 3	401	401	160273	161001.5	10760267	-0.2221	H_0
Test 4	401	401	161297	161001.5	10760267	0.09008	H_0
Test 5	401	401	159035	161001.5	10760267	-0.5995	H_0

As Table 5.3 shows, the values of the test statistic z^* for five tests were much smaller than 1.960 (the threshold), *i.e.* $|z^*| \leq 1.960$, the results were H_0 (meaning two distributions do not differ) at all the tests, indicating that the original and stego VoIP streams samples do not differ. The results suggest that the real-time steganographic system with embedded VoIP using the proposed steganographic algorithm is secure

in terms of statistical analysis, which has been widely used to assess the security of steganographic systems.

5.5.4 Algorithm Comparisons

The proposed steganographic algorithm uses hardware-based true random keys to protect the real-time steganographic system with embedded VoIP, whereas other related algorithms for VoIP steganographic systems use pseudorandom keys which are more vulnerable to cyber attacks.

For comparison purposes, the proposed steganographic algorithm and other related algorithms were used in the experiments to steganographically embed the secret message in VoIP streams, respectively.

Table 5. 4 Comparisons between the proposed algorithm and other related algorithms

Algorithm	Mean PESQ	Mean SNR	Data embedding rate (bps)
Proposed	4.21	44.87	796
Krätzer et al. (2006)	-	-	267
Huang et al. (2016)	3.33	-	442
Jiang et al. (2016)	4.02	44.33	500
Tian et al. (2017)	3.70	17.31	1700

Table 5.4 shows a comparison of the average PESQ, SNR and data embedding capacity (steganographic capacity) between the proposed steganographic algorithm

and other four steganographic algorithms (Tian et al., 2017; Huang et al., 2016; Krätzer et al., 2006; Jiang et al., 2016). For the proposed algorithm, twelve repeated experiments on real-time VoIP communications were carried out to determine the average measurements.

Close analysis shows, the real-time VoIP steganographic system with the proposed algorithm achieved the highest PESQ value in comparison with other related algorithms, and a slightly higher SNR value than Jiang's algorithm. These findings suggest that no or very little signal distortion occurs as a result of the proposed algorithm used in the real-time steganographic system with embedded VoIP.

The average data embedding rate of the proposed algorithm was much higher than those of Krätzer, Huang and Jiang's algorithms and was smaller than that of Tian's algorithm. The proposed algorithm achieved a higher PESQ value (4.21) and roughly half the embedding rate of Tian's algorithm, which is possible due to the fact that the latter made good use of redundancy in VoIP streams; however, the proposed algorithm uses hardware-based true random keys and various data embedding intervals in VoIP streams to steganographically embed the hidden message, thereby achieving much higher level of security than the latter.

Comparisons between the proposed steganographic algorithm and other related algorithms suggest that the proposed algorithm has great effectiveness in terms of speech quality and data embedding capacity, which are two of the important factors in designing a real-time steganographic system with embedded VoIP.

5.5.5 Security Analysis

In this study hardware-based true random keys are used to improve the security of the real-time steganographic system with embedded VoIP.

The security of the proposed algorithm is also based on the discrete logarithm problem: the problem of factoring large numbers. The problem in mathematics indicates the new algorithm has the advantage of being one of the most solid methods of secure VoIP communication over a public channel. It is generally believed that, to date, 663 bits long is the largest number factored by a general-purpose factoring algorithm using a state-of-the-art distributed implementation. Montgomery predicted that the next largest number should be 768 bits long (Montgomery, 2011). Although there are some doubts, it is reasonable to believe that 1024-bit keys may become breakable in the near term. Those full of optimism even feel that 4096-bit keys could be broken in the foreseeable future. Thus, it is presumed that a steganographic algorithm would be secure enough if the secret keys were sufficiently long, *e.g.* 1024 bits. In this study, all the keys used for steganographic VoIP communication are 1024 bits long, so the keys are unlikely to be broken in the near term, providing solid foundation for the real-time steganographic system with embedded VoIP.

5.6 Summary

A new steganographic algorithm using hardware-based true random keys is devised for real-time steganographic systems with embedded VoIP for secure communication in the research project. In the algorithm, true random number generation integrates with data embedding and data extraction. In addition, the read time stamp counter of

the CPU is used as a hardware entropy source to generate true random numbers as dynamic keys for real-time steganographic VoIP communications. The use of true random keys can ensure the secrecy of the secret message embedded in VoIP streams.

In the proposed algorithm, a series of random sequences generated from a logistic chaotic map are used to choose data embedding locations in VoIP streams. The use of chaotic map makes data embedding in VoIP streams randomly, so that it is unlikely to predict the initial conditions of random sequences. Thus the properties of the chaotic map can significantly increase the security of covert communications.

Experimental results and security analysis demonstrates the effectiveness of the proposed algorithm with imperceptible distortion of original VoIP signals and greater data embedding rates.

Statistical undetectability analysis using the M-W-W non-parameter test shows that the probability distributions drawn from the original VoIP streams and the stego VoIP streams do not differ for all sets of tests, indicating that the proposed steganographic algorithm is statistically undetectable with negligible signal distortion.

CHAPTER 6 Dynamic Key Distribution for Covert VoIP Communications

Voice over Internet Protocol (VoIP) is widely embedded into commercial and industrial applications. VoIP streams can be used as innocuous cover objects to hide secret data in steganographic systems. The security offered by VoIP signalling protocols is likely to be compromised due to a sharp increase in computing power. This chapter describes a theoretical and experimental investigation of covert steganographic communications over VoIP streams. A new theoretical model of steganographic VoIP communications was constructed to depict the security scenarios in steganographic systems against passive attacks. A one way accumulation-based steganographic algorithm was devised to integrate dynamic key updating and exchange with data embedding and extraction, so as to protect steganographic systems from adversary attacks. Theoretical analysis of steganographic security using information theory proves that the proposed model for covert VoIP communications is secure against a passive adversary. The effectiveness of the steganographic algorithm for covert VoIP communications was examined by means of performance and robustness measurements. The results reveal that the algorithm has no or little impact on real-time VoIP communications in terms of imperceptibility, speech quality and signal distortion, and is more secure and effective at improving the security of covert VoIP communications than other related algorithms with comparable data embedding rates.

6.1 Introduction

The past decade has witnessed the rapid development of embedded Voice over Internet Protocol (VoIP) for commercial and industrial applications. Widening access to the Internet greatly facilitates the use of multimedia applications in people's daily lives. Evolving network technology such as streaming has enjoyed a rise in popularity. However, security measures are struggling to keep up with the pace of change in attack tactics.

Encryption and decryption technologies are normally used to address data security and privacy issues. There are symmetric encryption and public-key encryption that enable the translation of a plaintext message into ciphertext. However, an increase in computing power has led to decryption of several encryption algorithms, such as MD5 (Wang et al., 2005), DES (EFF, 2016) and SHA-1 (Stevens et al, 2017), indicating possible vulnerabilities in the encryption primitives. It is generally recognised that encrypted messages are obvious, and when intercepted, it is clear that the communicating parties are communicating secretly.

As a sub-branch, digital steganography is defined as 'the art of concealed communication by hiding messages in seemingly innocuous objects' and 'the very existence of a steganographic message is secret' (Cox et al., 2008). Steganography in static cover objects, such as text, BMP or JPEG images, and WAV or MP3 audio files, has been explored extensively (Fridrich, 2014; Ker et al., 2014; Yang et al., 2019). Network protocols and streaming media (Stallings, 2017), such as VoIP, are also used to realise covert steganographic communications.

A great deal of research have been conducted on algorithm design and cover object selection for covert steganographic communications over streaming media, but little effort has been made to explore the potential of using dynamic key distribution to improve the security of steganographic systems.

Dittmann et al. first studied VoIP steganography and decryption techniques and suggested their algorithm (Dittmann et al., 2005). Aoki developed a lossless steganographic technique for G.711 telephony speech (Aoki, 2008), with the embedding capacity depending on the number of '0' in audio signals, so the practical application was limited. Liu et al. analysed the parameters of G.729 coded speech frames to identify the parameters and effective bits of G.729 speech coding, which were used for steganography (Liu et al., 2008). Yu et al. designed a VoIP steganographic scheme (Yu et al., 2009), but its validity needed to be confirmed. Aoki proposed a semi-lossless steganographic technique for G.711 telephony speech (Aoki, 2010), with bandwidth improved from 24 bit/s to 400 bit/s, depending on the background noise signal level. Huang et al. devised a high capacity steganographic algorithm for embedding data in various speech parameters of the inactive frames of low bit rate audio streams encoded with G.723.1 source codec (Huang et al., 2011). Tian et al. (2012) put forward a method to improve the performance of steganography by adding some similarity between the hidden message and the cover object to strike a balance between steganography transparency and bandwidth, but the similarity limited the choice of hidden messages. Gope et al. (2016) presented an authentication protocol for wireless sensors networks over which streaming media are transmitted; the protocol provided various imperative security properties such as user anonymity, untraceability, forward/backward secrecy, and perfect forward

secrecy. Tian et al. improved the security of quantization-index-modulation steganography in low bit-rate speech streams (Tian et al., 2014).

In 2016, Qi et al. used Discrete Spring Transform to eliminate redundancy in multimedia signals and improve speech quality (Qi et al., 2016). Liu et al. reported the use of a matrix embedding method to achieve steganography in linear predictive coding for low bit-rate speech codec (Liu et al., 2016). Janicki investigated pitch-based steganography using Speex voice codec (Janicki, 2016) to complement Aoki's work (2010). More recently, Tian et al. (2017) suggested a bitrate modulation steganographic algorithm with Hamming matrix encoding, but its practicality needed further study. Xin et al. proposed an adaptive audio steganographic algorithm for covert wireless communication, which was based on variable low bit coding (Xin et al., 2018). Overall, previous steganography studies mainly focused on steganographic algorithm design.

In summary, there has been a large body of research regarding steganographic algorithms for covert communications over streaming media, but the key distribution problem in covert steganographic communications has been sidestepped (Peng et al., 2020). In fact, the successfulness of steganographic algorithms for covert communications relies largely on the transmission of secret keys between the communicating parties. Security in transmission of secret keys is more crucial for covert VoIP communications because of the timing and loss of packets, i.e. covert VoIP communications require continuous embedding and necessary synchronization between the communicating parties. So far there are no reliable and secure key transmission schemes that could be put into use for covert VoIP communications. Thus, secure key transmission for covert steganographic communications is worth

studying apart from designing effective steganographic algorithms for them (Peng et al., 2020).

The main purpose of this chapter is to explore the potential of one way accumulation-based dynamic key updating and transmission for innovative applications in the field of covert steganographic communications over streaming media.

This chapter focuses on devising a new dynamic steganographic algorithm for covert VoIP communications. It includes one-way accumulation integrating into dynamic key updating and exchange, which can protect steganographic systems from man-in-the-middle attacks, which threaten covert steganographic communications.

The rest of this chapter is organised as follows: Section 6.2 describes key management and distribution which plays a vital role in VoIP communications. A novel theoretical model of steganographic VoIP communication is presented in Section 6.3. Section 6.4 details a new steganographic algorithm for covert VoIP communications. Security analysis of the new algorithm is discussed in Section 6.5. In Section 6.6, experiments including performance and security measurements are depicted, and the results are discussed in detail. The chapter is summarised in Section 6.7.

6.2 Key Management and Distribution

Public key cryptography provides solutions to the key distribution problem. A public key scheme such as RSA involves a public key and a private key; once the communicating parties compute the shared secret key they can use it as an encryption key (Hellman, 2002). The two keys are related due to inverse operations,

so there must be no easily computational method of deriving the private key from the public key. Public key schemes by themselves do not provide authentication of the communicating parties and are thus particularly vulnerable to man-in-the-middle attacks.

Key regression, time-evolving and multicast key distribution schemes were suggested for key management in cryptographic storage systems (Fu, 2006). They provided a means of deriving a sequence of temporally related keys from the most recent key. The effectiveness of these schemes for covert steganographic communications over streaming media is unknown since they are unable to address some key issues such as packet loss and synchronisation.

A number of authors have investigated key distribution schemes for image steganography. Dagar proposed an image steganographic algorithm that used two secret keys to randomise the bit hiding process and enhance the security of hidden messages (Dagar, 2014). Gutiérrez-Cárdenas suggested a PRNG key distribution scheme for image steganography, which used a picture to conceal a message with unaltered pixel information, so it could be secure against steganalysis detection (Gutiérrez-Cárdenas, 2014). Patel et al. reported LSB-based image steganography using dynamic key cryptography in which the dynamic feature of the key was enabled by rotating the key and each key rotation produced a new key (Patel et al., 2016). However, these key distribution schemes designed for image steganography cannot be used directly in covert steganographic communications over VoIP streams due to the timing and loss of packets.

Although the above investigations examined key distribution schemes for image steganography, to the best of the author's knowledge, only few references in the literature described key distribution schemes for VoIP steganography. This was the motivation behind the present study.

In comparison with existing steganographic algorithms for VoIP steganography, such as FIPlP (Jiang et al., 2016), CNV (Xiao et al., 2008), MELP (Krätzer et al., 2006; Dittmann et al., 2005), parameter-LSB (Tian et al., 2017) and HiF (Huang et al., 2016), the proposed steganographic algorithm is more secure for taking into account key distribution and authentication, and is more effective in terms of data embedding capacity and imperceptibility.

6.3 Steganographic VoIP Communication

A new theoretical model is devised for steganographic communications over VoIP streaming media in this research project. The model is based on steganography and cryptography and is depicted in Fig. 6.1 A secret message to be hidden (M) is encrypted with a secret key generated from a random number generator to form an encrypted message; the message is segmented into distinct parts which are embedded in a series of packets of media streams, namely cover objects (C). S in the figure denotes the packet containing a hidden message.

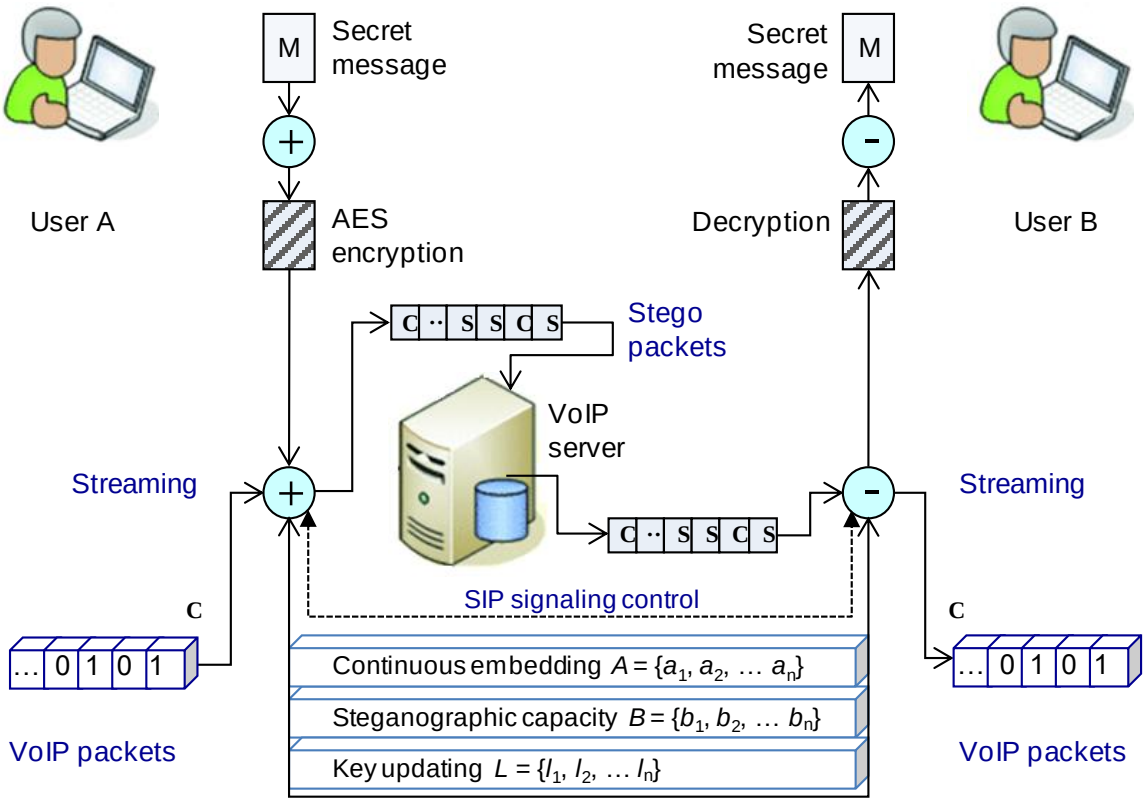

Figure 6. 1 Steganographic VoIP communications

This model integrates dynamic key distribution and authentication with data embedding and extraction. Three sequences in the model simulate the continuity of data embedding, time-variant size of hidden message in each packet and key pairs respectively, taking into account continuous data embedding and necessary synchronisation of sender and receiver due to packet loss.

The random sequence in Fig. 6.1, $A = \{a_1, a_2, \ldots a_n\}$, is a group of zeros, ones, twos and threes, e.g. {1,2,2,3,0,…}, describing the continuity of data embedding. The ones, twos and threes denote a packet containing the beginning, the continuation or the end of the hidden message, respectively, and zeros mean a packet does not contain the hidden message.

The sequence, $B = \{b_1, b_2, \ldots b_n\}$, is a set of steganographic capacity, corresponding to varying numbers of bits of the hidden message embedded in a series of streaming

media packets. This sequence enables the receiver to determine the size of the hidden message embedded in each packet.

The sequence, $L = \{l_1, l_2, \ldots l_n\}$, represents a set of private / public key pairs of 1024 bits each. The public and private keys are correlated in the public-key scheme where the sender and the receiver compute the shared key using knowledge of the public key based on discrete exponential and logarithm (hash) functions.

According to information theory, Kullback–Leibler (KL) divergence is used as a measure of security for steganographic systems (Fridrich, 2014).

The statistical distance (ε) between the cover object and the stego object can be expressed as $\varepsilon = \sum_{w \in W_0} P_C(w) - \sum_{w \notin W_0} P_C(w)$ and $W_0 \subset W$, where P_C is the probability distribution of the cover object, w is the measurement, W_0 is the plausible space, and W is the total space of possible measurements.

The total probability distribution of the secret message sent over the space W is given by

$$P_U(w) = P(w_i \in W_2)P_S(w) + P(w_i \in W_3)P_C(w) \quad (6.1)$$

where P_S is the probability distribution of the stego object, and W_2 and W_3 are the observation spaces relating to the stego object and the cover object, respectively. Equation (6.1) becomes $P_U(w) = \eta P_S(w) + (1-\eta)P_C(w)$, where η is the probability that '1' appears in a period. As

$$P_S(w) = \begin{cases} P_C(w)/1 + \varepsilon, & w \in W_0 \\ P_C(w)/1 - \varepsilon, & w \in W_1 \end{cases} \quad (6.2)$$

where W_0 and W_1 are the plausible spaces of possible measurements, then the relative entropy between the cover object and the stego object for covert steganographic communications is given by

$$D(P_C \| P_U) \leq \frac{1+\varepsilon}{2} \frac{\varepsilon^2 \eta^2}{1-\varepsilon^2(1-\eta)^2} \quad (6.3)$$

As Fig. 6.1 shows, there are two *N*-level true random sequences (*A* and *B*) used to model the dynamic and synchronisation characteristics of covert communications, $\eta = 2^{n-1}\, 2^{n-1}/(2^n - 1)(2^n - 1)$, then equation (6.3) becomes

$$D(P_C \| P_U) \leq \varepsilon^2 / (16 - 9\varepsilon^2) \leq \varepsilon \quad (6.4)$$

Hence, it proves the proposed model for covert communications over VoIP streaming media is secure against a passive adversary.

6.4 Dynamic Steganographic Algorithm for Covert VoIP Communication

In this research project, a novel dynamic steganographic algorithm is devised to integrate one way accumulation-based dynamic key distribution with data embedding and extraction for covert VoIP communications.

6.4.1 Accumulation-Based Key Distribution

The key updating and transmission algorithm is schematically shown in Fig. 6.2 for an illustrational purpose. The algorithm includes recurrences of Receiver validation (Step 1), Key transmission (Step 2), and Key updating (Step 3).

A one way cryptographic accumulation function is used to validate the communicating party, Bob, as shown in Fig. 6.2. Assuming $T = \{x_1, \ldots, x_n\}$ be the set of items$x_1, \ldots, x_n$

stored by Alice (the communicating party), she selects secure primes *p* and *q* that are suitably large, and a suitably large base *g* that is relatively prime to a big composite number *N*:

$$N = pq \tag{6.5}$$

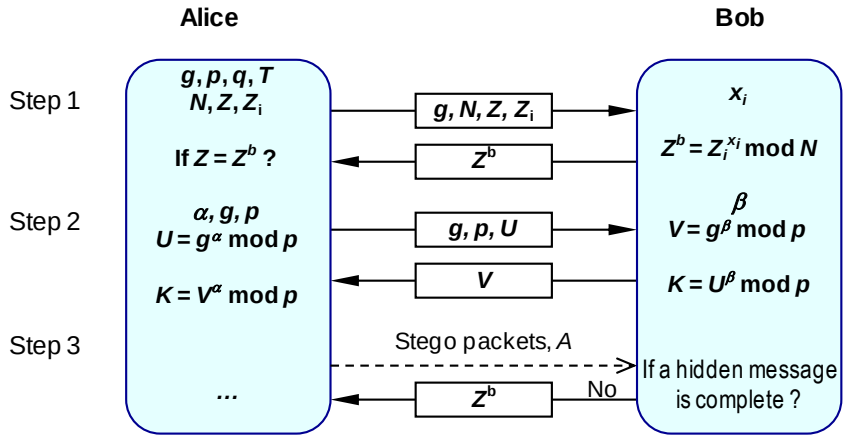

Figure 6. 2 Schematic description of key updating and transmission

The values of *g* and *N* are then made available to the public, but the values *p* and *q* are kept secret. Moreover, Alice computes the following value

$$Z = g^{x_1 x_2 \cdots x_n} \bmod N \tag{6.6}$$

and a partial accumulated hash value Z_i for Bob (x_i), *i.e.* the accumulation of all the values in the set *T* besides x_i.

$$Z_i = g^{x_1 x_2 \cdots x_{i-1} x_{i+1} \cdots x_n} \bmod N \tag{6.7}$$

Having computed Z_i, Alice sends Z_i and *N* to Bob, as well as the signed pair (*Z*, *t*) where *t* is the current timestamp. Bob determines whether *t* is current and (*Z, t*) is indeed signed by Alice. Bob then computes

$$Z^b = Z_i^{x_i} \bmod N \tag{6.8}$$

and returns the value of Z^b to Alice. Alice compares Z^b with *Z* that is calculated using equation (6.6). If the two agree, then Alice knows that the sender is Bob in possession of x_i. Indeed, it is generally accepted to be computational infeasible for someone who does not know the values of *p* and *q* to compute a value *Q* such that

$$Z = Q^{x_i} \bmod N \qquad (6.9)$$

when $x_i \notin T$, indicating that the one way accumulation function is secure in terms of cryptography.

After the receiver validation step, the private key used for encryption and decryption of a secret message is transmitted securely under Integer Decisional Diffie-Hellman assumption. Alice and Bob agree to use a prime number *p* and base *g* (*g* is assumed to be known by adversaries). Alice selects a secret integer α, calculates

$$U = g^{\alpha} \bmod p \qquad (6.10)$$

and sends the value of *U* to Bob. Bob chooses a secret integer β, computes

$$V = g^{\beta} \bmod p \qquad (6.11)$$

and sends the value of *V* to Alice. Alice calculates

$$(g^{\beta} \bmod p)^{\alpha} \bmod p$$

and Bob computes

$$(g^{\alpha} \bmod p)^{\beta} \bmod p$$

The values of $(g^{\beta} \bmod p)^{\alpha} \bmod p$ and $(g^{\alpha} \bmod p)^{\beta} \bmod p$ are the same because groups are power associative according to mathematical principles (Hellman, 2002). So both

Alice and Bob are now in possession of the group element $g^{\alpha\beta}$ mod *p*, which can serve as the shared secret key (*K*) to encrypt and decrypt the secret message:

$$K = (g^{\beta} \bmod p)^{\alpha} = (g^{\alpha} \bmod p)^{\beta} \qquad (6.12)$$

Key updating is the process of determining whether the stego packets received by Bob contain the complete hidden message. As Fig. 6.1 shows, the random sequence *A* is used to identify whether a packet contains the beginning, the continuation or the end of the hidden message. Using the sequence, Bob knows whether the secret message he decrypts with the shared key is complete or not.

To enable key exchange and transmission successfully in case of heavy packet loss, which is common in VoIP communications, the proposed algorithm contains a special re-distribution function as follows:

If a packet containing part of the secret message is lost, Bob sends Alice the value of Z^b again to initiate a repetition of Steps 1, 2 and 3, as shown in Fig. 6.2, *i.e.* Alice validates Bob as the 'legal' receiver (Step 1), and embeds the same secret message again (Step 2) until Bob receives all the packets used to embed the entire secret message (Step 3). Thus, this function can help achieve synchronization between the sender and the receiver, thereby eliminating the effect of packet loss on key exchange in covert steganographic communications over streaming media.

6.4.2 Data Embedding

Random number keys are used to encrypt the secret message to be hidden. The encrypted message is subsequently segmented into distinct parts, which are then

embedded in a series of packets of VoIP streams at different data embedding capacities and various data embedding locations. For simplicity, 16 bytes of VoIP streams have a data embedding capacity of two bytes. If the data embedding interval (*R*) is set to one, the secret message is randomly embedded in VoIP streams at an interval of two bytes, *i.e.* one byte of the secret message is randomly embedded in 16 bytes of VoIP streams. When the interval is set to two, the secret message is randomly embedded in VoIP streams at an interval of three bytes, so in this case 22 bytes of VoIP streams contain one byte of the secret message, and so on.

The process of embedding the secret message in VoIP streams is designed as follows:

Step A: first embed the secret message length (LoM) in VoIP streams, and set length to LoM.

Step B: embed the secret message into the first packet. Compute the length of the secret message embedded in the first packet (m_1) and the length of the secret message embedded in other packets (m_k).

Procedure DE First_packet

---

if (LoM < m_1)

then (

encrypt *M*(0, $m_0 - 1$) to form *E*(0, $m_0 - 1$)

embed *E*(0, $m_0 - 1$) in the bit stream *BIT* = {*bit*(0), *bit*(1), . . . , *bit*(($m_0 - 1$) *8)}

if (*bit*(i) == 0)

then (*V*(k) ←*V*(k)&0xfe)

else (*V*(k) ←*V*(k)|0x01)

k ← k + R

length ← 0

end

)

else (

encrypt $M(0, m_1 - 1)$ to form $E(0, m_1 - 1)$

embed $E(0, m_1 - 1)$ in the bit stream $BIT = \{bit(0), bit(1), \ldots, bit((m_1 - 1) * 8)\}$

if (*bit*(i) == 0)

then (*V*(k) ←*V*(k)&0xfe)

else (*V*(k) ←*V*(k)|0x01)

k ← k + R

length ← length - m_1

)

---

where m_0 is the total size of the secret message to be hidden, and k is the sequence number.

Step C: embed the secret message into the other packets.

Procedure DE Other_packets

---

while (length > m_k) **do**

(

encrypt $M(m_1, m_1 + m_k - 1)$ to form $E(m_1, m_1 + m_k - 1)$,

embed $E(m_1, m_1 + m_k - 1)$ in the bit stream $BIT = \{bit(0), bit(1), \ldots, bit((m_k - 1) * 8)\}$

if (*bit*(i) == 0)

then (*V*(k) ←*V*(k)&0xfe)

else (*V*(k) ←*V*(k)|0x01)

k ← k + R

length ← length - m_k

)

Step D: compute the length of the secret message embedded in the last packet (m_n)

Procedure DE Last_packet

---

encrypt *M*(LoM− length, LoM− 1) to form *E*(LoM− length, LoM− length + m_n − 1)

embed *E*(LoM−length, LoM−length+ m_n −1) into *BIT* = {*bit*(0), *bit*(1), . . . , *bit*((m_n − 1) * 8)}

if (*bit*(i) == 0)

then (*V*(k) ←*V*(k)&0xfe)

else (*V*(k) ←*V*(k)|0x01)

k ← k + R

length ← 0

end

For example, the first 16 bytes of the first VoIP packet is used to embed the length of the secret message to be hidden, and the remaining of the first packet and the other VoIP packets are used to embed the secret message itself. The size of the remaining of the first packet is about 4080 bytes, with a data embedding capacity up to 510 bytes (12.5%). To reduce the encryption time, the first 496 bytes of the secret message are embedded in the first VoIP packet. AES is used to encrypt the first 496 bytes of the secret message, and the resulting ciphertext is then embedded in VoIP packets using the data embedding algorithm above. As for the other VoIP packets, 512 bytes of the secret message (encrypted with AES) are embedded in each VoIP packet. The remaining of the secret message is embedded in the last VoIP packet with the size of LLoM. The LLoM value may not be a multiple of 16 bytes, so it is possibly necessary to adjust it to a multiple of 16 bytes in some cases.

6.4.3 Data Extraction

The extraction of the hidden secret message, steganographically embedded in VoIP streams using the data embedding algorithm above, from the stego VoIP streams is the inverse process of the data embedding algorithm described in the previous section. The corresponding extraction algorithm is used to retrieve the secret message encrypted with AES, and decrypt it with the same secret keys to obtain the original secret message from stego VoIP packets.

6.5 Security Analysis

This section theoretically examines the security of the new steganographic algorithm for covert VoIP communications, and shows how the algorithm can resist possible adversary attacks, which threaten existing VoIP steganographic algorithms.

6.5.1 Authentication for Communicating Parties

Covert channels based on one way accumulation are used to conduct key updating and transmission for covert steganographic communications over VoIP streams.

The general form of a cryptographic accumulator can be defined as follows: first start with a 'seed' value y_0, denoting the empty set, then define the accumulation value incrementally from y_0 for a set of elements $T = \{x_1, \ldots, x_n\}$, so that $y_i = f(y_{i-1}, x_i)$, where f is a one way function whose final value does not depend on the order of the x_i's (Goodrich et al., 2002). So a source can digitally sign the value of y_n in order to enable

a third party to produce a short proof for any element x_i belonging to T – namely, swap x_i with x_n and recompute y_{n-1} from scratch – the pair (x_i, y_{n-1}) is a cryptographically-secure assertion for the membership of x_i in the set T.

A new key distribution algorithm is devised in this research project to offer secure key updating and transmission for covert steganographic communications over streaming media, i.e. a one way cryptographic accumulator along with Diffie-Hellman key exchange are integrated with data embedding during steganography. The use of the accumulator provides cryptographic authentication for the communicating parties, thereby preventing covert steganographic communications from adversary attacks.

6.5.2 Man-in-the-Middle Attacks

The integration of a new one way cryptographic accumulator and Diffie-Hellman based key exchange is used to provide secure key exchange for covert steganographic communications over streaming media. The algorithm can ensure secure key updating and transmission, and then protect steganographic systems from adversary attacks.

Dynamic key distribution in the algorithm means the keys are almost unlikely to be compromised, because it enables steganographic systems to continually and randomly generate new private keys that the communicating parties share automatically. As the private key is changed continuously, a compromised key in the system could only decrypt a small amount of encoded information using today's supercomputers.

The man-in-the-middle attack is defined as a form of active eavesdropping in which

an attacker makes independent connections with the communicating parties and relays messages between them, making the communicating parties believe that they are talking directly to each other over a connection; in fact the entire conversation is controlled by the attacker (Stallings, 2017).

In the new steganographic algorithm, a one way cryptography accumulator is used to conduct authentication between the communicating parties to prevent possible man-in-middle attacks. As described in Section 6.4, Alice authenticates Bob (*i.e.* Bob in possession of a valid x_i) by determining whether Z^b (computed using equation (6.8) and sent to Alice by Bob) is equal to Z (computed by Alice using equation (6.6)). If the third party John wants to launch a man-in-the-middle attack, he has to pass the verification process first. Obviously without the knowledge of the two primes of p and q, John cannot pass the verification process of the receiver. In addition, as John cannot guess the high entropy element x_i correctly, he cannot work out the value of Z^b equal to Z. Thus, John cannot launch the man-in-the-middle attack successfully to cheat Alice, indicating that the proposed steganographic algorithm can prevent man-in-the-middle attacks.

6.5.3 Adversary Attacks

As for covert VoIP communications, the essential security is that it would not cause any suspicion from adversaries. Once a secret communication is suspicious, or an attacker has noticed that there is an underlying communications channel, the whole covert communications system is not safe, because the attacker can intercept and even destroy the communications.

The steganographic security follows the same path as security in cryptography (Fridrich, 2014). The security of covert steganographic communication lies in the fact that nobody has so far been able to produce an attack substantially faster than brute-force search for the key.

In the proposed steganographic algorithm, the Diffie–Hellman key exchange scheme (Hellman, 2002) is used to securely exchange the initial parameters of the chaotic map between the communicating parties that are authenticated using a one way accumulation-based authentication protocol devised in Section 6.4.1, instead of conventional elliptic curve digital signatures. As discussed in Section 6.5.2, the use of authentication with the one way accumulation authentication can prevent man-in-the-middle attacks, which are particularly possible on wireless networks.

However, there are limitations on resisting tampering attacks. Mutual authentication could be used to prevent tampering attacks, and it is a good solution to resist tampering attacks by sending an authentication message of the secret message to the receiver. If the verification fails on the receiver side, the secret message would be retransmitted. Although the utilization of Message Authentication Code (MAC) could resist the tampering attacks, but the computation of MAC is costing, which would add latency and lead to speech distortion – it may not be acceptable as to real-time VoIP communications over the Internet. Besides, MAC is a kind of redundant message which would reduce the available embedding capacity. Thus, it is difficult to achieve security and efficiency simultaneously in real-time VoIP communications with steganography.

6.6 Experimental Results and Discussion

This section summarises the findings, interprets what the steganographic communications results mean, and explains the significance of the results.

VoIP experiments were used to evaluate the performance and security of the proposed dynamic key distribution-based steganographic algorithm for covert communications over streaming media. Performance measurements were carried out using Digital Speech Level Analyser (DSLA) (DSLA, 2013), as shown in Fig. 6.3, which determines ITU-T P.862 objective speech quality scoring plus improved Mean Opinion Score (MOS) prediction according to ITU-T P.862.1 (ITU-T Recommendation, 2001). The experiments were repeated 12 times to assess the effectiveness and security of the new steganographic algorithm.

In the experiments, VoIP streams with PCM format encoded with G.711 codec were used as cover objects for covert steganographic communications over streaming media. The secret message to be hidden was encrypted with random number keys, and the encrypted message was divided into segments that were then embedded in a series of packets of VoIP streams. The imperceptibility of the resulting stego VoIP streams was evaluated, and the data embedding capacity was computed accordingly for each set of experiments.

The ITU-T P.862 recommendation (ITU-T Recommendation, 2001) was adopted to measure the subjective qualities of the stego VoIP streams. The recommendation describes an objective method for predicting the subjective quality of narrowband speech codecs. It uses the perceptual evaluation speech quality (PESQ) value and the signal-to-noise ratio (SNR) to assess the subjective quality of the stego VoIP

streams. The DSLA, made by Malden Electronics Ltd, UK (DSLA, 2013), was used in the experiments for PESQ and SNR measurements, as shown in Fig. 6.3. DSLA is a professional measurement device for objective speech quality prediction and speech level measurement in telecommunications equipment and networks. DSLA performs ITU-T P.862 (PESQ) objective speech quality scoring plus improved MOS prediction according to ITU-T P.862.1 as well as Wideband P.862.2 score.

PESQ score is calculated according to P.862 and PESQ P.862.1 gives a quality score on a MOS-like scale for narrowband listening. The aim of the amended recommendation ITU-T P.862.1 is to provide a single mapping from the raw P.862 score to the Listening Quality Objective Mean Opinion Score (MOS-LQO). The mapping from PESQ score to PESQ P.862.1 is performed as follows:

$$PESQ\text{P.862.1} = 0.999 + \frac{4.999 - 0.999}{1 + e^{-1.4945 \times PESQScore + 4.6607}} \quad (6.13)$$

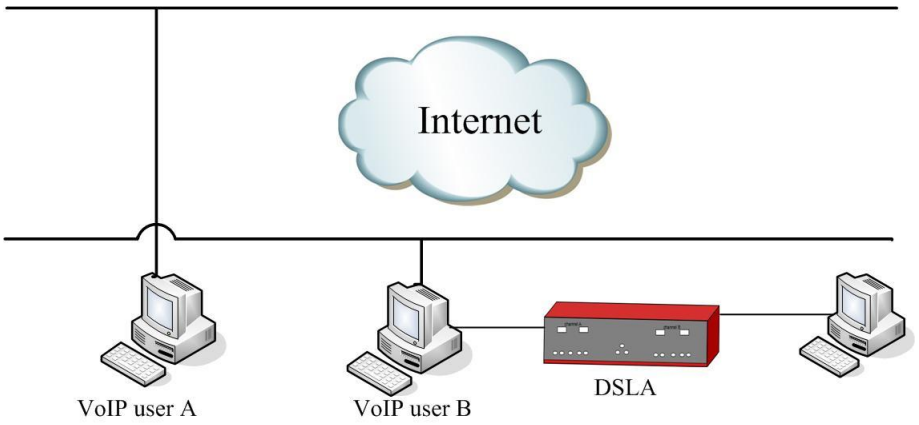

Figure 6. 3 Diagram of measurements for covert VoIP communications

6.6.1 Imperceptibility

The essential security of covert VoIP communication is that it would not cause any suspicion from attackers. Experiments were carried out to determine the degree of imperceptibility of steganographic systems using the proposed algorithm.

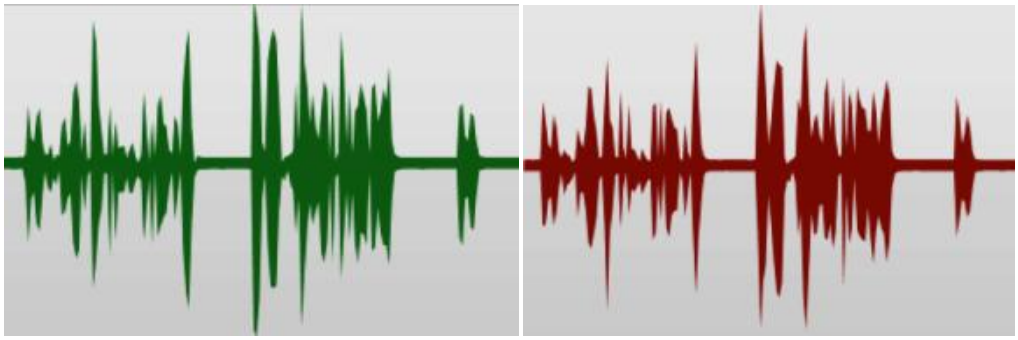

Figure 6. 4 Waveforms of the original and stego VoIP streams

Figure 6.4 shows the waveforms of the original VoIP streams (left image) and the stego VoIP streams that contain a secret message (right image), respectively. As Fig. 6.4 shows, there was a remarkable resemblance between the two waveforms. Listening tests indicated that a human perceptual system could not distinguish the differences between the original VoIP streams and the stego VoIP streams with the hidden message. The results show very little distortion occurred in the time domain as a result of steganography in VoIP streams.

Figure 6.5 shows a comparison of the mean PESQ values for the original VoIP streams, the stego VoIP streams with AES embedding-based steganography, and the stego VoIP streams using the proposed steganographic algorithm. For each testing, 12 repeated experiments were carried out to yield the mean PESQ value. The multiple-line graph demonstrates that the PESQ values were reasonably stable except for Testing 8 in which the PESQ was close to 3.5 (the lower boundary of good speech quality). This is in line with the expectation that the new steganographic

algorithm would cause no or little degradation in speech quality whilst improving the security of steganographic systems by means of dynamic key distribution, indicative of secure and robust covert VoIP communications.

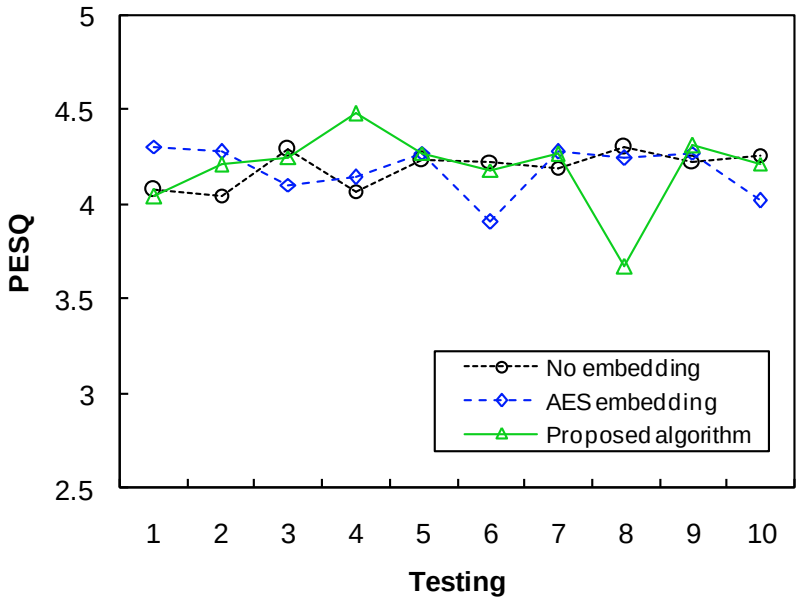

Figure 6. 5 PESQ values for the original VoIP streams and the stego VoIP streams with AES steganography or the proposed algorithm

Figure 6.6 shows a comparison of the mean SNR values between the original VoIP streams, the stego VoIP streams with AES embedding-based steganography, and the stego VoIP streams using the proposed steganographic algorithm. The SNR variance of the original streams was estimated to be 3.1, that of the stego streams with AES steganography was around 5.5, and that of the stego streams using the new algorithm was around 4.6. These results indicate that covert VoIP communications using the new steganographic algorithm have much better imperceptibility with a greater level of security than AES steganography.

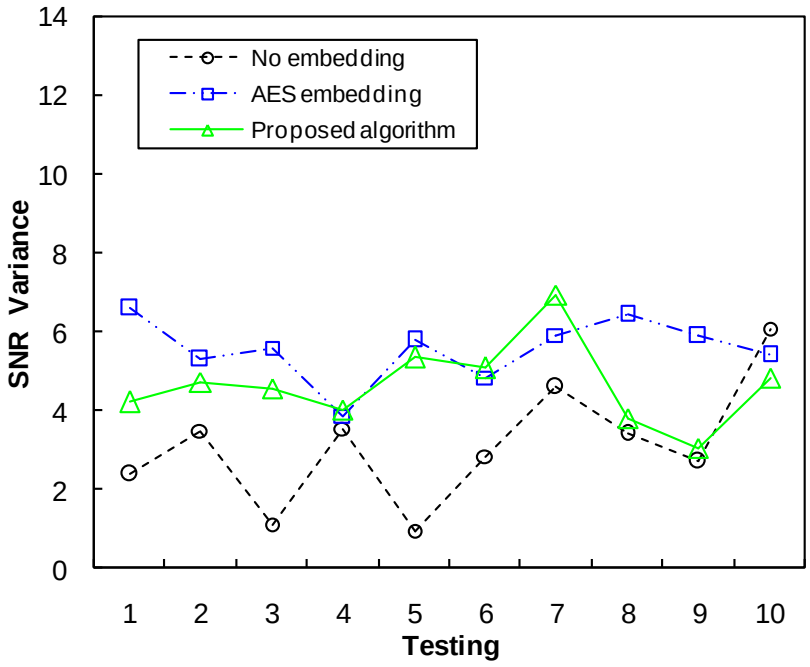

Figure 6. 6 SNR values for the original VoIP streams and the stego VoIP streams with AES steganography or the proposed algorithm

Overall, the results above indicate that the proposed steganographic algorithm has no or little impact on real-time VoIP communications in terms of speech quality, signal distortion and imperceptibility. The differences in PESQ and SNR between the original and stego VoIP streams were so minor that distortion resulted from covert communications using the new algorithm was imperceptible, indicating that the proposed steganographic algorithm is effective at breaking through the key exchange bottleneck occurs in covert steganographic communications over streaming media, and protecting steganographic systems from man-in-the-middle attacks.

6.6.2 Effects of Data Embedding Intervals

The secret message to be hidden is embedded in a series of packets of VoIP streams at various data embedding intervals, so as to study the effects of the increased

complexity of the proposed steganographic algorithm on covert steganographic communications over streaming media.

Figure 6.7 shows changes in the mean PESQ values of the stego VoIP streams that contain the hidden message encrypted with AES before embedding at different data embedding interval distances in streaming media. As Fig. 6.7 shows, the average PESQ values of the stego streams decreased slightly before the data embedding interval distance reached 3, and then increased as the interval distance increased, showing an upward trend gradually close to the mean PESQ value of the VoIP streams without data embedding (4.31). The results indicate the advantage of the increased complexity of the proposed steganographic algorithm.

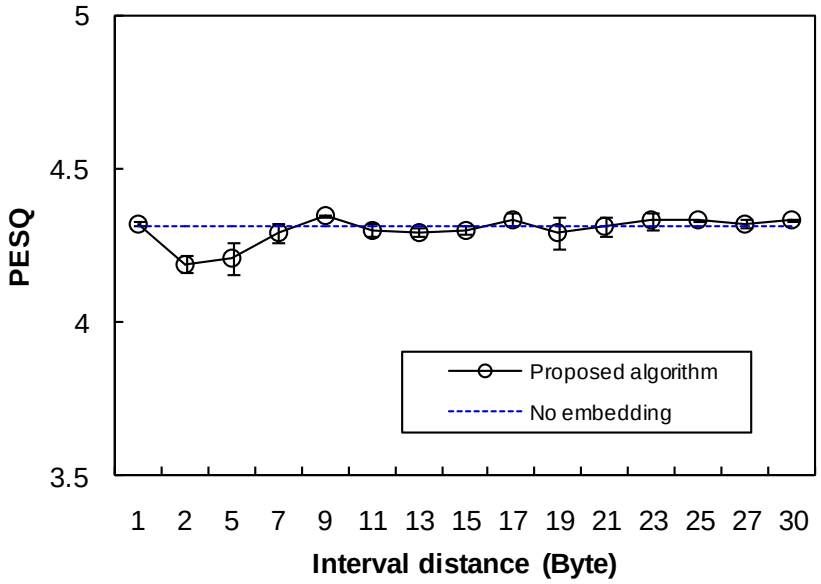

Figure 6. 7 PESQ values of the stego VoIP streams at various interval distances

Figure 6.8 shows changes in mean SNR values of the stego VoIP streams at various data embedding interval distances in comparison to that of the original VoIP streams. Each SNR value is the average of the SNR values of 12 repeated experiments. The average SNR value of the original streams was measured to be around 48.83 decibels. The SNR values of the stego streams decreased slightly and then increased

as the interval distance reached 7 bytes. With the interval distance increasing, the SNR levelled off with small fluctuations, and the average SNR value was 45.81 decibels approximately. The SNR variance between the original and stego streams was 3.02 decibels, i.e. 6.18% change in SNR after steganography, indicating that the proposed steganographic algorithm has a negligible impact on real-time VoIP communications, regardless of data embedding intervals.

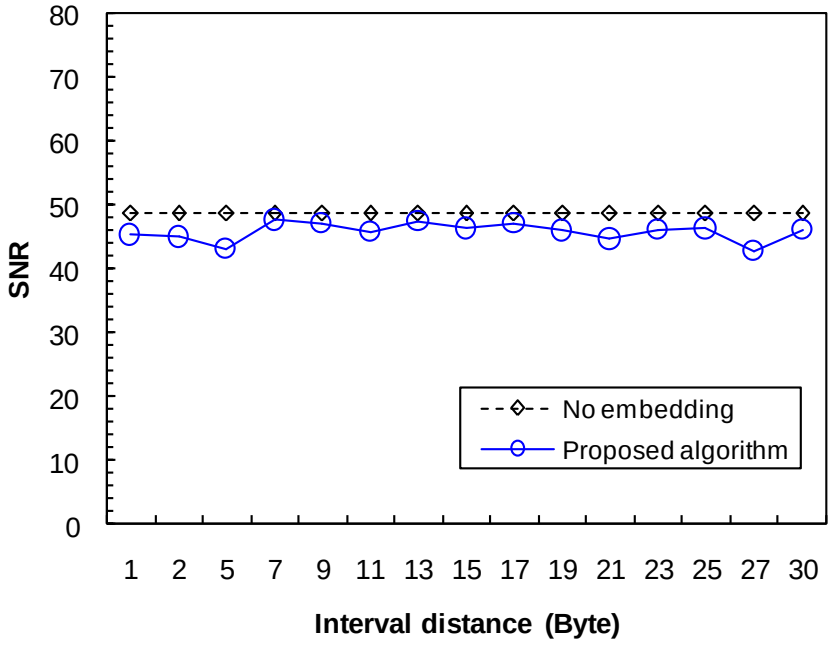

Figure 6. 8 SNR values of VoIP streams at various interval distances.

6.6.3 Effects of Hidden Message Size

To study the effects of the size of the hidden message on the data embedding capacity of the proposed steganographic algorithm, a series of covert VoIP communications experiments were carried out.

Figure 6.9 shows changes in the average PESQ values of the stego VoIP streams taken from covert VoIP communications using the new steganographic algorithm at

different sizes of the hidden message. Each data point is the mean value based on 12 repeated experiments. As Fig. 6.9 shows, the average PESQ values of the stego streams decreased with the hidden message size increasing. The average PESQ values of the stego VoIP streams were still greater than 3.5, the lower threshold of covert VoIP communications for the codec used in the experiments, before the hidden message size reached 1186 bytes, which can be regarded as the maximum data embedding capacity. When the size of hidden message exceeds the embedding capacity of the covert object, speech quality would decrease seriously, leading to unavailability of real-time covert communication.

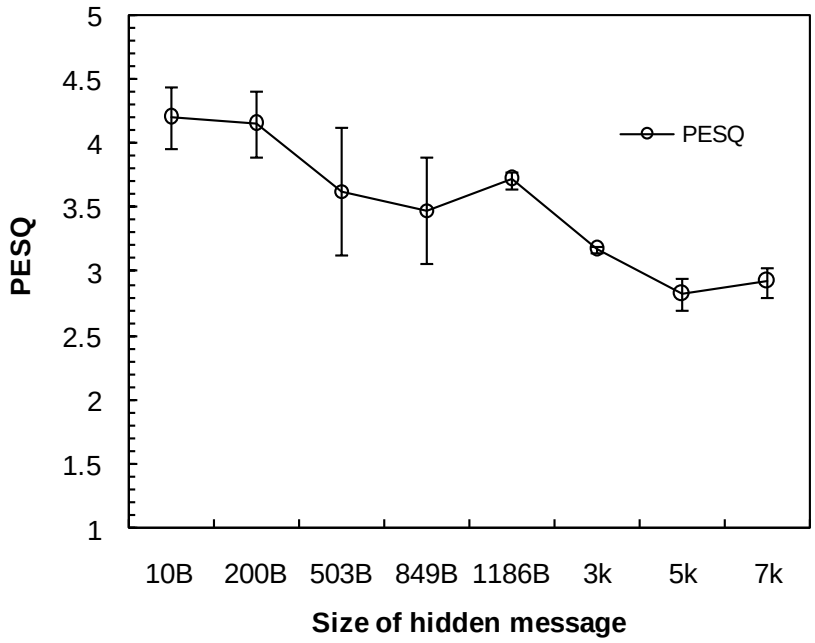

Figure 6. 9 PESQ values of the stego VoIP streams varying with the hidden message size.

The SNR measurements were conducted accordingly in the experiments. Figure 6.10 shows change in the average SNR values of the original VoIP streams and the stego VoIP streams using the proposed steganographic algorithm at different sizes of the hidden messages. The SNRs of the stego streams decreased with the hidden

message size increasing. There was a steady decrease in the SNR variance when the hidden message size increased. It is generally recognized that a higher SNR would be better, meaning less distortion. The results suggest that the proposed steganographic algorithm causes little signal distortion when the hidden message size is not greater than 1186 bytes.

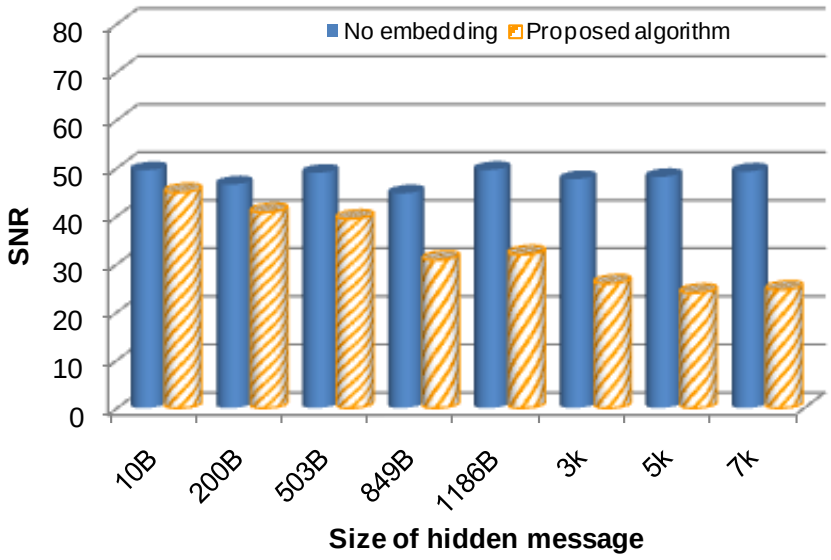

Figure 6. 10 SNR values of the original VoIP streams and the stego VoIP streams using the proposed algorithm at various hidden message sizes.

6.6.4 Statistical Undetectability Analysis

Steganographic communications aim to conceal the existence of hidden messages from both human perceptual systems and computational detection. Statistical undetectability is normally used to evaluate the security of a steganographic system (Fridrich, 2014). A secure steganographic system should be statistically undetectable.

In order to evaluate the security of the proposed steganographic algorithm, t-test was used in this research project to perform statistical undetectability analysis. The t-test

is a statistical hypothesis test in which the test statistic follows a Student's t-distribution under the null hypothesis (Box, 1987). A two-sample t-test is used when it can be assumed that two distributions have the same variance. The *t* statistic used to test whether the difference between the two samples is significant can be calculated as follows:

$$t = \frac{\overline{X_1} - \overline{X_2}}{\sqrt{\frac{(n_1 - 1)S_1^2 + (n_2 - 1)S_2^2}{n_1 + n_2 - 2}(\frac{1}{n_1} + \frac{1}{n_2})}} \quad (6.14)$$

where S_1^2 and S_2^2 are the variances of the samples;n_1 and n_2 are the sample sizes.

The default alpha of 0.05 is normally used as the threshold. When the calculated *P*-value is less than the threshold, there is a significant difference between the two samples.

Table 6. 1 Undetectability analysis results using t-test

	Original stream samples	Stego stream samples
Mean	-3.25151E-07	5.30214E-06
Variance	0.000918239	0.000919384
Observations	79872	79872
P(*T*<=*t*) *two-tail*	0.970405697	

The original and stego VoIP stream samples were tested in our experiments. Among the variables that appear in Table 6.1, the *P*-value is 0.970405697, which is greater than 0.05, indicating no significant difference between the tested samples. The results show that the covert communication system using the proposed steganographic algorithm is secure in terms of statistical undetectability analysis.

6.6.5 Comparisons with Other Related Algorithms

To confirm the effectiveness of the proposed steganographic algorithm, comparisons of the data embedding capacity, number of communication passes, message size for authentication, collision resistance, computational overhead and bandwidth between the proposed algorithm and other related algorithms were conducted for covert steganographic communications over VoIP streams.

Some steganographic algorithms, such as FIPIP (Jiang et al., 2016), CNV (Xiao et al., 2008), MELP (Krätzer et al., 2006; Dittmann et al., 2005), parameter-LSB (Tian et al., 2017), and HiF (Huang et al., 2016), have been suggested for VoIP steganography. These algorithms achieved different levels of data embedding in streaming media. For comparison purposes, these existing algorithms and the proposed steganographic algorithm were used in the experiments to steganographically embed the secret message in VoIP streams, respectively.

Figure 6.11 shows a comparison of the average data embedding capacity between the proposed steganographic algorithm and other four steganographic algorithms. For each algorithm, 12 repeated experiments on covert VoIP communications were carried out to determine the average data embedding rate. As Fig. 6.11 shows, the average data embedding rate of the proposed algorithm was much higher than those of the MELP and CNV algorithms, approximately equal to that of the parameter-LSB algorithm, and lagged behind the HiF algorithm. The average data embedding rate of the proposed algorithm was lower than that of the HiF algorithm, which is possible due to the fact that the HiF algorithm made good use of redundancy in the inactive

frames of VoIP streams; however, the proposed algorithm uses various data embedding intervals in VoIP streams to steganographically embed the secret message, thereby achieving much higher level of security than the HiF algorithm. The data embedding results suggest that the proposed steganographic algorithm has great effectiveness in terms of the average data embedding rate, which is sometimes one of the important factors in designing a steganographic algorithm for covert steganographic communications over streaming media.

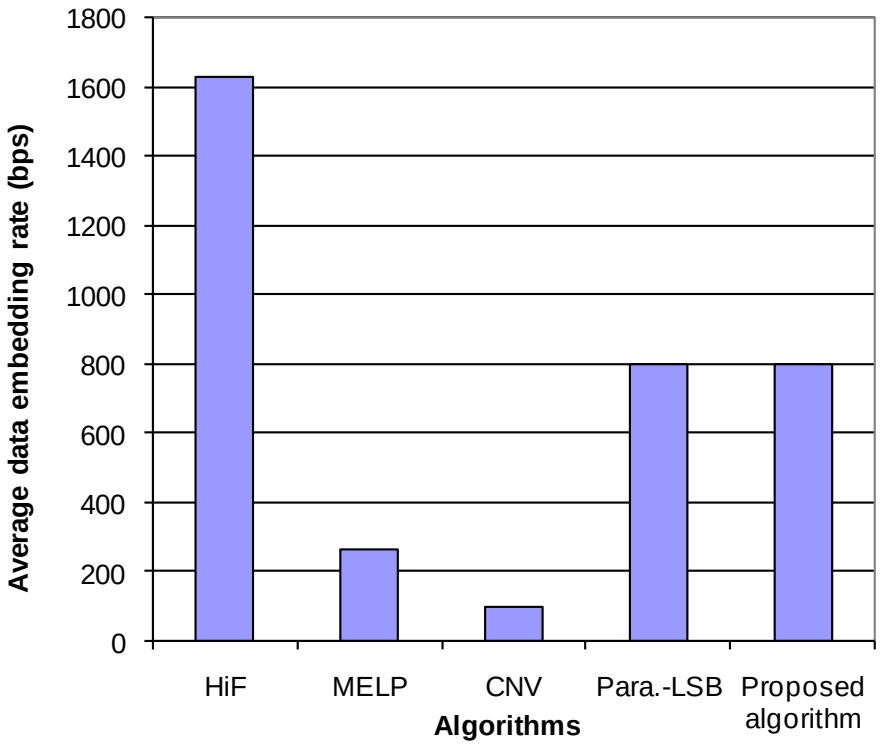

Figure 6. 11 Comparison of data embedding rates between the proposed algorithm and other related algorithms.

Table 6.2 shows a comparison of the number of required communication passes between the proposed steganographic algorithm and other related algorithms. The 'N/A' means the value is not available to the public. As the table shows, the number of required communication passes of the proposed algorithm is 10 times, which is

higher than FIPlP, HiF and MELP, providing cryptographic authentication for covert communication systems.

Table 6. 2 Comparison of the Number of Required Communication Passes

Algorithm	Number of required passes
Proposed	10
FIPlP	5
HiF	5
CNV	N/A
MELP	4
parameter-LSB	N/A

Table 6.3 shows a comparison of required message sizes for authentication and collision resistance between the proposed algorithm and other related algorithms. As for the proposed covert communication system and FIPlP, the size of audio data in each packet for authentication is 4096 bytes. The strength of steganographic systems against brute-force attacks depends on the block length for key construction and the key size. With a birthday attack (a typical cryptographic attack), it is possible to find a collision of an *n*-bit key in $2^{n/2}$. MELP used MD5 (128-bit) to produce a shorter hash value to calculate a checksum for the hidden message, with a collision of $2^{128/2}$. As Table 6.3 shows, a birthday attack on the proposed system produces a collision with a work factor of approximately $2^{1152/2}$, which is viewed as adequate to provide sufficient collision resistance as to today's computing power.

Table 6. 3 Comparison of Required Message Size for Authentication and Collision Resistance

Algorithm	Required message size	Collision resistance
Proposed	4096 bytes	$2^{1152/2}$
FIPlP	4096 bytes	$2^{192/2}$
HiF	N/A	$2^{128/2}$
CNV	N/A	N/A
MELP	N/A	$2^{128/2}$
parameter-LSB	N/A	N/A

Table 6.4 shows a comparison of computational overhead and bandwidth between the proposed algorithm and other related algorithms. The computational overhead of the proposed algorithm is 88.45 ms, which is acceptable for real-time covert VoIP communication. As with MELP, the computational overhead value is the average of those estimated at four sampling rates used. The parameter-LSB algorithm had the highest steganographic bandwidth due to the use of less secure, simple LSB, and the bandwidth values for other five algorithms were comparable. As can be seen from Table 6.4 and Figs. 6.5 and 6.6, the proposed algorithm achieved a relatively larger steganographic bandwidth (0.80 kbits/s) with negligible signal distortion, which is one of the most important performance metrics that assess covert communication over streaming media.

Table 6. 4 Comparison of Computational Overhead and Bandwidth

Algorithm	Computational overhead (ms)	Bandwidth (kbits/s)
Proposed	88.45	0.80
FIPlP	N/A	0.50
HiF	N/A	0.44
CNV	240	0.10
MELP	2760	0.43
parameter-LSB	N/A	1.70

As for covert VoIP communication, it is very unlikely that an adversary will be able to obtain many different copies of a given stego VoIP packet due to real-time, dynamic and streaming features; therefore, collusion attacks are of less or no concern. That means the covert VoIP communication system has incredible strength and great resilience considering collusion attacks. Security analysis of the new algorithm for covert VoIP communication is detailed in Section 6.5.

6.7 Summary

The purpose of the current study was to explore the potential of one way accumulation-based dynamic key distribution for innovative applications in the field of covert steganographic communications. The new steganographic algorithm devised for covert steganographic communications over VoIP streams was found to be secure against a passive adversary. The evidence from the study suggests that the algorithm can protect steganographic systems from adversary attacks such as man-in-the-middle attacks. Security analysis and experimental results show the

effectiveness of the proposed algorithm with imperceptible distortion of the original signals (5% change in PESQ) and greater data embedding rates (~ 800 bps). The findings from this study add to a growing body of literature on steganography in streaming media.

CHAPTER 7 Conclusions and Future Perspectives

7.1 Overview

Voice over Internet Protocol (VoIP) is widely embedded into commercial and industrial applications. VoIP streams can be used as innocuous cover objects to hide secret data in steganographic systems. The security offered by VoIP signalling protocols is likely to be compromised due to a sharp increase in computing power. This research project addresses these security issues systematically.

This thesis embodies the methods and results of the research project undertaken by the PhD candidate over the last three years. It provides new insights into how streaming steganography and advanced cryptography such as advanced encryption standard and dynamic key distribution can work together to substantially improve the security, performance and robustness of steganographic systems, which can act as a covert VoIP communications channel to protect the secret data embedded in streaming packets during VoIP communications.

7.2 Research Findings and Innovations

This thesis has answered the research questions stated in Chapter 1, thereby addressing the problems of theoretical model, uncertainty of the embedding rate of media packets and integrity of the secret message. It has met the objectives as

follows:

A. Develop an information theoretical model for covert steganographic communication over streaming media

Chapter 4 addresses the first research question, i.e. how does the new information theoretical model of secure covert communications over streaming media depict the security scenarios in streaming media-based steganographic systems with passive attacks.

The information theoretical model of secure covert communications over streaming media devised in this project was shown to be suitable for depicting the security scenarios in streaming media-based steganographic systems with passive attacks. An information source for covert VoIP communications using streaming media steganography was modelled using a stochastic process with greater precision. The theory of hypothesis testing was discovered to be effective at analysing the adversary's detection performance. A discrete prediction model of high precision was capable of simulating the characteristics of time-variance of streaming payloads in covert communications over streaming media. Theoretical analysis of steganographic security using information theory proved that the proposed model for covert VoIP communications is secure against a passive adversary.

B. Devise a new algorithm that uses hardware as the entropy source to generate true random numbers as dynamic keys to be used by covert VoIP communications

Chapter 5 addresses the second research question, i.e. how does hardware-based technology generate true random numbers as secret keys for covert VoIP communication using digital steganography to ensure the security of cryptographic

systems.

The potential of hardware-based true random key generation for innovative applications in a real-time steganographic system with embedded VoIP for streaming communication was examined. The hardware random number generator that used the read time stamp counter of the CPU as an entropy source was proved to be effective in generating true random numbers as dynamic keys for the new steganographic algorithm devised in this study for VoIP communication systems to protect data hidden in real-time VoIP streams.

Data embedding locations in VoIP streams were chosen randomly according to random sequences generated from a logistic chaotic map designed in this project. The use of a key-distribution scheme to exchange the initial parameters of the chaotic map between the VoIP users was discovered to be not affecting and disrupting the functioning of the SIP signalling protocol used to establish connections between the communicating parties in the signalling phase in real-time VoIP communications.

C. Devise a secure dynamic key agreement and updating algorithm based on One-way accumulation, which is applicable to covert VoIP communications.

Chapter 6 addresses the third research question, i.e. how does a one-way cryptographic accumulator work along with dynamic key updating and transmission, which is integrated with the data embedding and extraction processes in covert VoIP communications.

Streaming steganography complementing advanced cryptography technologies, such as true random key generation, advanced encryption standard and dynamic key

distribution and management, was found to be an effective way to substantially improve the security, performance and robustness of steganographic systems, as indicated by comparison of experimental data and theoretical expectation of the speech metrics of VoIP streams.

Non-parameter statistical tests, security analysis and performance experiments, including PESQ, SNR and data embedding capacity, were conducted to compare the proposed steganographic algorithm with other related algorithms. The results show that the steganographic VoIP system achieved the mean PESQ of 4.21, the mean SNR of 44.87, and the average data embedding capacity up to 796 bps, in the presence of moderate packet loss, indicating that the new algorithm is effective at improving the security of steganographic systems with embedded VoIP.

The one way accumulation-based steganographic algorithm devised in this work was clearly suitable for integrating dynamic key updating and exchange with data embedding and extraction, so as to protect steganographic systems from adversary attacks. The effectiveness of the steganographic algorithm for covert VoIP communications was examined by means of performance and robustness measurements and statistical undetectability analysis. The results reveal that the algorithm has no or little impact on real-time VoIP communications in terms of imperceptibility, speech quality and signal distortion, and is more secure and effective at improving the security of covert VoIP communications than other related algorithms with comparable data embedding rates.

D. Perform undetectability analysis and steganalysis of covert VoIP communications using the Mann-Whitney-Wilcoxon test and t-test.

Part of Chapter 5 and Chapter 6 addresses the last research question, i.e. how can a statistical test be used to analyse the adversary's detection performance.

Non parameter Mann-Whitney-Wilcoxon statistical testing, detailed in Chapter 5, and T-test, depicted in Chapter 6, were found to be an effective way to analyse the adversary's detection performance on distinguishing between an innocent cover object and a modified stego object containing a hidden message, thus proving the effectiveness of the new steganographic algorithms devised in this PhD research.

The findings from the study make several contributions to the current literature as follows:

a) A novel information theoretical model of steganographic VoIP communication is constructed to realise secure covert VoIP communications, achieving high data embedding capacities comparable to other related algorithms.

b) A new dynamic steganographic algorithm is devised for covert VoIP communications. It includes one way accumulation integrating into dynamic key updating and exchange, which can protect steganographic systems from man-in-the-middle attacks, which threaten covert steganographic communications.

c) A secure real-time VoIP system, underpinned by a novel steganographic algorithm using a logistic chaotic map to maximise algorithm complexity, is devised to enable the system to be embedded into complex industrial systems without compromising security.

d) A hardware generator that uses the read time stamp counter of the CPU is developed to generate true random numbers as dynamic keys for VoIP steganographic systems. These true random keys can ensure the secrecy of the

steganographic message.

e) Performance evaluation of the secure real-time VoIP system takes place by means of state-of-the-art network equipment Digital Speech Level Analyser, unlike previous works with performance evaluation being conducted using in-house software with low precision.

f) Security analysis is carried out using the Mann-Whitney-Wilcoxon test and t-test, instead of conventional statistical tests.

7.3 Research Limitations

According to information theory, the steganographic security follows the same path as security in cryptography. The security of covert steganographic communication lies in the fact that nobody has so far been able to produce an attack substantially faster than brute-force search for the key.

The dynamic steganographic algorithm devised in this research project uses a one way accumulation-based authentication protocol to authenticate the communicating parties on a VoIP network. The use of authentication with the accumulative authentication can prevent man-in-the-middle attacks, which are particularly possible on wireless networks, as well as collision attacks.

The limitation of the steganographic VoIP system developed in the work is the lack of the mechanism for resisting tampering attacks. Message Authentication Code (MAC) can be used to resist tampering attacks, but execution time is so long that it will lead to latency and possible speech distortion, which is not acceptable as to real-time VoIP

communications over the Internet. In addition, MAC is a redundant message which would reduce available data embedding rates. It is therefore difficult to achieve security and efficiency simultaneously for real-time VoIP communications with steganography.

7.4 Future Research

Classical steganography theory cannot be applied directly to video steganographic systems, since video payloads are highly dynamic and subject to both image and audio-induced signal distortion.

The information theoretical model of secure covert communications over streaming media developed in this research project could enable the investigation of video steganography, if the model were slightly modified to allow image and audio-induced distortion to be taken into account, thus giving important information on the security metrics as a function of the message size and / or data embedding rate. The information gained would be equally useful in addressing video steganography security issues.

The dynamic steganographic algorithm for covert VoIP communications devised in this research could be extended to depict the security scenarios in streaming video-based steganographic systems with passive attacks. This could be achieved by integrating the VoIP steganographic algorithm with an image steganographic algorithm to take account of image and audio-induced signal distortion.

The mutual authentication technique designed in this work needs to be further improved, to study the prevention of tampering attacks in the presence of heavy

packet loss. It should be a good solution to resist tampering attacks by sending an authentication message of the secret message to the receiver over a VoIP network. If the verification failed on the receiver side, the secret message would be retransmitted until authentication takes place.

References

Ahsan, K., & Kundur, D. (2002, December). Practical data hiding in TCP/IP. In Proc. Workshop on Multimedia Security at ACM Multimedia (Vol. 2, No. 7, pp. 1-8).

Aoki, N. (2004, November). VoIP packet loss concealment based on two-side pitch waveform replication technique using steganography. In 2004 IEEE Region 10 Conference TENCON 2004. (Vol. 100, pp. 52-55). IEEE.

Aoki, N. (2007, November). Potential of value-added speech communications by using steganography. In Third International Conference on Intelligent Information Hiding and Multimedia Signal Processing (IIH-MSP 2007) (Vol. 2, pp. 251-254). IEEE.

Aoki, N. (2008, August). A technique of lossless steganography for G. 711 telephony speech. In 2008 International Conference on Intelligent Information Hiding and Multimedia Signal Processing (pp. 608-611). IEEE.

Aoki, N. (2009, August). Lossless steganography techniques for IP telephony speech taking account of the redundancy of folded binary code. In 2009 Fifth International Joint Conference on INC, IMS and IDC (pp. 1689-1692). IEEE.

Aoki, N. (2010, October). A semi-lossless steganography technique for G. 711 telephony speech. In 2010 Sixth International Conference on Intelligent Information Hiding and Multimedia Signal Processing (pp. 534-537). IEEE.

Arackaparambil, C., Yan, G., Bratus, S., & Caglayan, A. (2012, January). On tuning the knobs of distribution-based methods for detecting VoIP covert channels. In 2012 45th Hawaii International Conference on System Sciences (pp. 2431-2440). IEEE.

Argyris, A., Deligiannidis, S., Pikasis, E., Bogris, A., & Syvridis, D. (2010). Implementation of 140 Gb/s true random bit generator based on a chaotic photonic integrated circuit. Optics express, 18(18), 18763-18768.

Bai, L. Y., Huang, Y., Hou, G., & Xiao, B. (2008, August). Covert channels based on jitter field of the RTCP header. In 2008 International Conference on Intelligent Information Hiding and Multimedia Signal Processing (pp. 1388-1391). IEEE.

Bayon, P., Bossuet, L., Aubert, A., Fischer, V., Poucheret, F., Robisson, B., & Maurine, P. (2012, May). Contactless electromagnetic active attack on ring oscillator based true random number generator. In International Workshop on

Constructive Side-Channel Analysis and Secure Design (pp. 151-166). Springer, Berlin, Heidelberg.

Box, J. F. (1987). Guinness, gosset, fisher, and small samples. Statistical science, 45-52.

Brassard, G. (1984, December). Quantum cryptography: public key distribution and coin tossing int. In Conf. on Computers, Systems and Signal Processing (Bangalore, India (pp. 175-9).

Cachin, C. (1998, April). An information-theoretic model for steganography. In International Workshop on Information Hiding (pp. 306-318). Springer, Berlin, Heidelberg.

Chen, S., Wang, X., & Jajodia, S. (2006). On the anonymity and traceability of peer-to-peer VoIP calls. IEEE Network, 20(5), 32-37.

Cheng, Q., & Huang, T. S. (2001). An additive approach to transform-domain information hiding and optimum detection structure. IEEE Transactions on Multimedia, 3(3), 273-284.

Chikha, R. J. B., Abbes, T., Chikha, W. B., & Bouhoula, A. (2016). Behavior-based approach to detect spam over IP telephony attacks. International Journal of Information Security, 15(2), 131-143.

Cisco. “Internetworking Technology Handbook.” 2003. URL: http://www.cisco.com/univercd/cc/td/doc/cisintwk/ito_doc/ (26 October 2004)

Collier, M. (2004). The value of VoIP security.

Cover, T. M., Thomas, J. A., & Kieffer, J. (1994). Elements of information theory. SIAM Review, 36(3), 509-510.

Cox, I. J., Matthew, L., & Miller, J. A. (2008). Bloom, Jessica ridrich and Ton Kalker," Digital Watermarking and Steganography".

Cox, I., Miller, M., Bloom, J., Fridrich, J., & Kalker, T. (2007). Digital watermarking and steganography. Morgan kaufmann.

Daemen, J., & Rijmen, V. (2003). AES Proposal: Rijndael National Institute of Standards and Technology.

Dagar, S. (2014, May). Highly randomized image steganography using secret keys. In International Conference on Recent Advances and Innovations in Engineering (ICRAIE-2014) (pp. 1-5). IEEE.

Danger, J. L., Guilley, S., & Hoogvorst, P. (2009). High speed true random number generator based on open loop structures in FPGAs. Microelectronics journal,

40(11), 1650-1656.

Deng, Z., Shao, X., Yang, Z., & Zheng, B. (2008). A novel covert speech communication system and its implementation. Journal of Electronics (China), 25(6), 737-745.

Diffie, W., & Hellman, M. (1976). New directions in cryptography. IEEE transactions on Information Theory, 22(6), 644-654. [8] R.L. Rivest, A. Shamir, and L. Adleman, “On Digital Signatures and Public Key Cryptosystems,” Commun. Ass. Comp. Mach., vol. 21, pp. 120-126, 1978.

Dittmann, J., Hesse, D., & Hillert, R. (2005, March). Steganography and steganalysis in voice-over IP scenarios: operational aspects and first experiences with a new steganalysis tool set. In Security, Steganography, and Watermarking of Multimedia Contents VII (Vol. 5681, pp. 607-618). International Society for Optics and Photonics.

DRUID. 2007. Real-time steganography with RTP. Technical Report

DSLA II Getting Started Guide Revision 3.0, Malden Electronics Ltd, UK, 2013.

EFF. Frequently Asked Questions (FAQ) About the “EFF DES Cracker” Machine, 2016.

European project, SECOQC White Paper on Quantum Key Distribution and Cryptography (www.secoqc.net/downloads/secoqc_crypto_wp.pdf).

FIPS, P. (2001). 197: Advanced encryption standard (AES). National Institute of Standards and Technology, 26.

Forbes, C. (2009). A new covert channel over RTP.

Fridrich, J. (2014). Steganography in digital media: principles, algorithms, and applications, 2nd ed. Cambridge University Press.

Fu, K., Kamara, S., & Kohno, T. (2006). Key regression: Enabling efficient key distribution for secure distributed storage. Computer Science Department Faculty Publication Series, 149.

Geiser, B., Mertz, F., & Vary, P. (2008, October). Steganographic Packet Loss Concealment forWireless VoIP. In ITG Conference on Voice Communication [8. ITG-Fachtagung] (pp. 1-4). VDE.

Goode, B. (2002). Voice over internet protocol (VoIP). Proceedings of the IEEE, 90(9), 1495-1517.

Goodrich, M. T., Tamassia, R., & Hasić, J. (2002, September). An efficient dynamic and distributed cryptographic accumulator. In International Conference on

Information Security (pp. 372-388). Springer, Berlin, Heidelberg.

Gope, P., & Hwang, T. (2016). A realistic lightweight anonymous authentication protocol for securing real-time application data access in wireless sensor networks. IEEE Transactions on industrial electronics, 63(11), 7124-7132.

Greenstreet, D., & Scoggins, S. (2005). Building Residential VoIP Gateways: A Tutorial Part Four: VoIP Security Implementation. Building Residential VoIP Gateways: A Tutorial. URL: http://www. analogzone. com/nett0913. pdf (26 October 2004).

Gutiérrez-Cárdenas, J. M. (2014, July). Secret Key Steganography with Message Obfuscation by Pseudo-random Number Generators. In 2014 IEEE 38th International Computer Software and Applications Conference Workshops (pp. 164-168). IEEE.

Hamdaqa, M., & Tahvildari, L. (2011, June). ReLACK: a reliable VoIP steganography approach. In 2011 Fifth International Conference on Secure Software Integration and Reliability Improvement (pp. 189-197). IEEE.

Hellman, M. E. (2002). An overview of public key cryptography. IEEE Communications Magazine, 40(5), 42-49.

Hirabayashi, M., Kojima, H., & Oiwa, K. (2010). Design of true random one-time pads in DNA XOR cryptosystem. In Natural Computing (pp. 174-183). Springer, Tokyo.

Huang, Y. F., Tang, S., & Yuan, J. (2011). Steganography in inactive frames of VoIP streams encoded by source codec. IEEE Transactions on information forensics and security, 6(2), 296-306.

Huang, Y., & Tang, S. (2016). Covert voice over internet protocol communications based on spatial model. Science China Technological Sciences, 59(1), 117-127.

Huang, Y., Liu, C., Tang, S., & Bai, S. (2012). Steganography integration into a low-bit rate speech codec. IEEE transactions on information forensics and security, 7(6), 1865-1875.

Huang, Y., Xiao, B., & Xiao, H. (2008, August). Implementation of covert communication based on steganography. In 2008 International Conference on Intelligent Information Hiding and Multimedia Signal Processing (pp. 1512-1515). IEEE.

Huang, Y., Yuan, J., Chen, M., & Xiao, B. (2011). Key distribution over the covert communication based on VoIP. Chinese Journal of Electronics, 20(2), 357-360.

Hussain, S. U., Majzoobi, M., & Koushanfar, F. (2016). A built-in-self-test scheme for online evaluation of physical unclonable functions and true random number

generators. IEEE Transactions on Multi-Scale Computing Systems, 2(1), 2-16.

International Engineering Consortium. “H.323.” 2004. URL: https://www.techonline.com/directory/international-engineering-consortium-iec.

James F. Kurose, Keith W. Ross. (2000). Computer Networking--A Top_Down Approach. 6th Edition, 672-688.

Janicki, A. (2016). Pitch‐based steganography for Speex voice codec. Security and communication networks, 9(15), 2923-2933.

Janicki, A., Mazurczyk, W., & Szczypiorski, K. (2015). Influence of speech codecs selection on transcoding steganography. Telecommunication Systems, 59(3), 305-315.

Jiang, R., Zhou, H., Zhang, W., & Yu, N. (2017). Reversible data hiding in encrypted three-dimensional mesh models. IEEE Transactions on Multimedia, 20(1), 55-67.

Jiang, Y., Tang, S., Zhang, L., Xiong, M., & Yip, Y. J. (2016). Covert voice over Internet protocol communications with packet loss based on fractal interpolation. ACM Transactions on Multimedia Computing, Communications, and Applications (TOMM), 12(4), 1-20.

Jie Liu, W., Chen, H. W., Li, Z. G., Liu, Z. H., & Xiao, F. Y. (2008, June). Efficient quantum secure direct communication with authentication. In 2008 IEEE Congress on Evolutionary Computation (IEEE World Congress on Computational Intelligence) (pp. 1764-1768). IEEE.

Jun, B., & Kocher, P. (1999). The Intel random number generator. Cryptography Research Inc. white paper, 27, 1-8.

Ker, A. D., & Pevný, T. (2014). The steganographer is the outlier: Realistic large-scale steganalysis. IEEE Transactions on information forensics and security, 9(9), 1424-1435.

Komaki, N., Aoki, N., & Yamamoto, T. (2003). A packet loss concealment technique for VoIP using steganography. IEICE TRANSACTIONS on Fundamentals of Electronics, Communications and Computer Sciences, 86(8), 2069-2072.

Kratzer, C., Dittmann, J., Vogel, T., & Hillert, R. (2006, May). Design and evaluation of steganography for voice-over-IP. In 2006 IEEE International Symposium on Circuits and Systems (pp. 4-pp). IEEE.

Kuhn, D. R., Walsh, T. J., & Fries, S. (2005). Security considerations for voice over IP systems. NIST special publication, 800.

Kurose, J. F., & Ross, K. W. Computer networking: a top-down approach (pp. 607967-5). Addison Wesley.

Li, S. B., Tao, H. Z., & Huang, Y. F. (2012). Detection of quantization index modulation steganography in G. 723.1 bit stream based on quantization index sequence analysis. Journal of Zhejiang University SCIENCE C, 13(8), 624-634.

Li, S., & Ephremides, A. (2005, March). A covert channel in MAC protocols based on splitting algorithms. In IEEE Wireless Communications and Networking Conference, 2005 (Vol. 2, pp. 1168-1173). IEEE.

Lin, Y. T., Wang, C. M., Chen, W. S., Lin, F. P., & Lin, W. (2016). A novel data hiding algorithm for high dynamic range images. IEEE Transactions on Multimedia, 19(1), 196-211.

Liu, J., Zhou, K., & Tian, H. (2012, June). Least-significant-digit steganography in low bitrate speech. In 2012 IEEE International Conference on Communications (ICC) (pp. 1133-1137). IEEE.

Liu, L., Li, M., Li, Q., & Liang, Y. (2008, August). Perceptually transparent information hiding in G. 729 bitstream. In 2008 International Conference on Intelligent Information Hiding and Multimedia Signal Processing (pp. 406-409). IEEE.

Liu, P., Li, S., & Wang, H. (2017). Steganography integrated into linear predictive coding for low bit-rate speech codec. Multimedia Tools and Applications, 76(2), 2837-2859.

Lloyd, P. (2010). An exploration of covert channels within Voice over IP.

Ma, L., Wu, Z., & Yang, W. (2007, August). Approach to hide secret speech information in G. 721 scheme. In International Conference on Intelligent Computing (pp. 1315-1324). Springer, Berlin, Heidelberg.

Martin, K. (2007, June). Steganographic communication with quantum information. In International Workshop on Information Hiding (pp. 32-49). Springer, Berlin, Heidelberg.

Mazurczyk, W. (2012). Lost audio packets steganography: the first practical evaluation. Security and Communication Networks, 5(12), 1394-1403.

Mazurczyk, W., & Kotulski, Z. (2006a, September). New VoIP traffic security scheme with digital watermarking. In International Conference on Computer Safety, Reliability, and Security (pp. 170-181). Springer, Berlin, Heidelberg.

Mazurczyk, W., & Kotulski, Z. (2006b). New security and control protocol for VoIP based on steganography and digital watermarking. arXiv preprint cs/0602042.

Mazurczyk, W., & Lubacz, J. (2010). LACK—a VoIP steganographic method. Telecommunication Systems, 45(2-3), 153-163.

Mazurczyk, W., & Szczypiorski, K. (2008, June). Covert Channels in SIP for VoIP signalling. In International Conference on Global e-Security (pp. 65-72). Springer, Berlin, Heidelberg.

Mazurczyk, W., & Szczypiorski, K. (2008, November). Steganography of VoIP streams. In OTM Confederated International Conferences" On the Move to Meaningful Internet Systems" (pp. 1001-1018). Springer, Berlin, Heidelberg.

Mazurczyk, W., Szaga, P., & Szczypiorski, K. (2014). Using transcoding for hidden communication in IP telephony. Multimedia Tools and Applications, 70(3), 2139-2165.

McEliece, R. J. (1978). A public-key system based on algebraic coding theory, 114-116. Deep sace network progress report, 44. Jet Propulsion Laboratory, California Institute of Technology.

Mehta, P. C., & Udani, S. (2001). Overview of Voice over IP.

Miao, R., & Huang, Y. (2011, June). An approach of covert communication based on the adaptive steganography scheme on Voice over IP. In 2011 IEEE International Conference on Communications (ICC) (pp. 1-5). IEEE.

Montgomery, P. L. (2008, October). Preliminary Design of Post Sieving Processing for RSA 768. In CADO workshop on integer factorization (October 2008).

Murdoch, S. J., & Lewis, S. (2005, June). Embedding covert channels into TCP/IP. In International Workshop on Information Hiding (pp. 247-261). Springer, Berlin, Heidelberg.

Nagireddi, S. (2008). VoIP voice and fax signal processing. John Wiley & Sons.

Neter, J., Wasserman, W., & Whitmore, G. A. (1993). Applied statistics. 4th ed. Simon & Schuster

Nishimura, A. (2009). Steganographic band width extension for the AMR codec of low-bit-rate modes. In Tenth Annual Conference of the International Speech Communication Association.

NIST. “Voice Over Internet Protocol (VoIP), Security Technical Implementation Guide.” Version 1, Release 1. 13 January 2004. URL: http://csrc.nist.gov/pcig/STIGs/VoIP-STIG-V1R1R-4PDF.pdf (26 October 2004)

Nutzinger, M., & Wurzer, J. (2011, August). A novel phase coding technique for steganography in auditive media. In 2011 Sixth International Conference on Availability, Reliability and Security (pp. 91-98). IEEE.

Nutzinger, M., Fabian, C., & Marschalek, M. (2010, October). Secure hybrid spread spectrum system for steganography in auditive media. In 2010 Sixth International

Conference on Intelligent Information Hiding and Multimedia Signal Processing (pp. 78-81). IEEE.

Patel, N., & Meena, S. (2016, November). LSB based image steganography using dynamic key cryptography. In 2016 International Conference on Emerging Trends in Communication Technologies (ETCT) (pp. 1-5). IEEE.

Pazarci, M., & Dipcin, V. (2003, July). Data embedding in scrambled digital video. In Proceedings of the Eighth IEEE Symposium on Computers and Communications. ISCC 2003 (pp. 498-503). IEEE.

Peng, J. & Tang, S. (2020). Covert communication over VoIP streaming media with dynamic key distribution and authentication. IEEE Transactions on Industrial Electronics, in Press (doi: 10.1109/TIE.2020.2979567).

Peng, J., Tang, S., & Jia, L. (2019). Fast Fourier Transform-based steganalysis of covert communications over streaming media. International Journal of Computer and Information Engineering, 13(7), 362-367.

Qi, Q., Peng, D., & Sharif, H. (2016). DST approach to enhance audio quality on lost audio packet steganography. EURASIP Journal on Information Security, 2016(1), 20.

Qiu, Q. (2003). Study of digest authentication for Session Initiation protocol (SIP). SITE, University of Ottawa, Ontario, Canada.

Qu, H., Su, P., & Feng, D. (2004, October). A typical noisy covert channel in the IP protocol. In 38th Annual 2004 International Carnahan Conference on Security Technology, 2004. (pp. 189-192). IEEE.

Ramsdell, B. (Ed.). (1999). RFC2633: S/MIME Version 3 Message Specification.

Recommendation, I. T. (2001). Perceptual evaluation of speech quality (PESQ): An objective method for end-to-end speech quality assessment of narrow-band telephone networks and speech codecs. Rec. ITU-T P. 862.

Rizal, M. (2014). A Study of VoIP performance in anonymous network-The onion routing (Tor) (Doctoral dissertation, Niedersächsische Staats-und Universitätsbibliothek Göttingen).

Rosenberg, J., Schulzrinne, H., Camarillo, G., Johnston, A., Peterson, J., Sparks, R., ... & Schooler, E. (2002). SIP: Session initiation protocol.

Sellke, S. H., Wang, C. C., Bagchi, S., & Shroff, N. (2009, April). TCP/IP timing channels: Theory to implementation. In IEEE INFOCOM 2009 (pp. 2204-2212). IEEE.

Shah, G., Molina, A., & Blaze, M. (2006, July). Keyboards and Covert Channels. In

USENIX Security Symposium (Vol. 15). [64] SHAH G., BLAZE M. 2009. Covert channels through external interference, In Proc. of the 3rd USENIXconference on Offensive technologies, Montreal, Canada

Shannon, C. E. (1949). Communication theory of secrecy systems. Bell system technical journal, 28(4), 656-715.

Simmons, G. J. (1983). ªThe Prisoners' Problem and the Subliminal Channel, Advances in Cryptology. In Proc. CRYPTO (Vol. 83, pp. 51-67).

Speech Quality Assessment, Malden Electronics Ltd, UK, 2007.

Stallings, W. (2017). Cryptography and Network Security: Principles and Practice. 7th Edition, 29-32, 86-140,252-279, 426-500.

Stallings, W. and Brown, L. (2018). Computer Security: Principles and Practice. 4th Edition. London: Pearson.

Stallings, W., Brown, L., Bauer, M. D., & Bhattacharjee, A. K. (2019). Computer security: principles and practice 4th Edition (pp. 77-79). London: Pearson

Standards for Security Categorization of Federal Information and Information Systems [FIPS 199].

Stevens, M., Bursztein, E., Karpman, P., Albertini, A., & Markov, Y. (2017, August). The first collision for full SHA-1. In Annual International Cryptology Conference (pp. 570-596). Springer, Cham.

Takahashi, T., & Lee, W. (2007, September). An assessment of VoIP covert channel threats. In 2007 Third International Conference on Security and Privacy in Communications Networks and the Workshops-SecureComm 2007 (pp. 371-380). IEEE.

Tang, S., Chen, Q., Zhang, W., & Huang, Y. (2016). Universal steganography model for low bit‐rate speech codec. Security and Communication Networks, 9(8), 747-754.

Tang, S., Jiang, Y., Zhang, L., & Zhou, Z. (2014). Audio steganography with AES for real-time covert voice over internet protocol communications. Science China Information Sciences, 57(3), 1-14.

The NIST Computer Security Handbook [NIST95].

Tian, H., Guo, R., Lu, J., & Chen, Y. (2012). Implementing covert communication over voice conversations with windows live messenger. Advances in Information Sciences and Service Sciences, 4(4).

Tian, H., Jiang, H., Zhou, K., & Feng, D. (2011). Adaptive partial-matching

steganography for voice over IP using triple M sequences. Computer Communications, 34(18), 2236-2247.

Tian, H., Jiang, H., Zhou, K., & Feng, D. (2012). Transparency-orientated encoding strategies for voice-over-IP steganography. The Computer Journal, 55(6), 702-716.

Tian, H., Liu, J., & Li, S. (2014). Improving security of quantization-index-modulation steganography in low bit-rate speech streams. Multimedia systems, 20(2), 143-154.

Tian, H., Sun, J., Chang, C. C., Qin, J., & Chen, Y. (2017). Hiding information into voice-over-IP streams using adaptive bitrate modulation. IEEE Communications Letters, 21(4), 749-752.

Tian, H., Zhou, K., & Feng, D. (2010). Dynamic matrix encoding strategy for voice-over-IP steganography. Journal of Central South University of Technology, 17(6), 1285-1292.

Tian, H., Zhou, K., Huang, Y., Feng, D., & Liu, J. (2008, November). A covert communication model based on least significant bits steganography in voice over IP. In 2008 The 9th International Conference for Young Computer Scientists (pp. 647-652). IEEE.

Tian, H., Zhou, K., Jiang, H., Huang, Y., Liu, J., & Feng, D. (2009b, May). An adaptive steganography scheme for voice over IP. In 2009 IEEE International Symposium on Circuits and Systems (pp. 2922-2925). IEEE.

Tian, H., Zhou, K., Jiang, H., Liu, J., Huang, Y., & Feng, D. (2009a, June). An M-sequence based steganography model for voice over IP. In 2009 IEEE International Conference on Communications (pp. 1-5). IEEE.

Tucker, G. S. (2005). Voice Over Internet Protocol (VoIP) and Security. white paper, SANS Institute.

Veljković, F., Rožić, V., & Verbauwhede, I. (2012, March). Low-cost implementations of on-the-fly tests for random number generators. In 2012 Design, Automation & Test in Europe Conference & Exhibition (DATE) (pp. 959-964). IEEE.

Vera-del-Campo, J., Guasch, S., Pegueroles, J., & Soriano, M. (2015). Private surveys on VoIP. Computer Communications, 68, 25-32.

Wang, C., & Wu, Q. (2007, December). Information hiding in real-time VoIP streams. In Ninth IEEE International Symposium on Multimedia (ISM 2007) (pp. 255-262). IEEE.

Wang, X., & Yu, H. (2005, May). How to break MD5 and other hash functions. In Annual international conference on the theory and applications of cryptographic techniques (pp. 19-35). Springer, Berlin, Heidelberg.

Wang, X., Chen, S., & Jajodia, S. (2005, November). Tracking anonymous peer-to-peer voip calls on the internet. In Proceedings of the 12th ACM conference on Computer and communications security (pp. 81-91).

Wieser, C., & Röning, J. (2011). An evaluation of VoIP covert channels in an SBC setting. SECURITY IN FUTURES–SECURITY IN CHANGE, 54.

Wiesner, S. (1983). Conjugate coding. ACM Sigact News, 15(1), 78-88.

Wu, Z. J., Wei, G. A. O., & Wei, Y. A. N. G. (2009). LPC parameters substitution for speech information hiding. The Journal of China Universities of Posts and Telecommunications, 16(6), 103-112.

Wu, Z. J., Yang, W., & Yang, Y. X. (2003). ABS-based speech information hiding approach. Electronics Letters, 39(22), 1617-1619.

Wu, Z., & Yang, W. (2006, August). G. 711-based adaptive speech information hiding approach. In International Conference on Intelligent Computing (pp. 1139-1144). Springer, Berlin, Heidelberg.

Xiao, B., Huang, Y., & Tang, S. (2008, November). An approach to information hiding in low bit-rate speech stream. In IEEE GLOBECOM 2008-2008 IEEE Global Telecommunications Conference (pp. 1-5). IEEE.

Xin, G., Liu, Y., Yang, T., & Cao, Y. (2018). An adaptive audio steganography for covert wireless communication. Security and Communication Networks, 2018.

Xu, E., Liu, B., Xu, L., Wei, Z., Zhao, B., & Su, J. (2011, September). Adaptive VoIP steganography for information hiding within network audio streams. In 2011 14th International Conference on Network-Based Information Systems (pp. 612-617). IEEE.

Xu, T., & Yang, Z. (2009, November). Simple and effective speech steganography in G. 723.1 low-rate codes. In 2009 International Conference on Wireless Communications & Signal Processing (pp. 1-4). IEEE.

Yang, B., Rožić, V., Mentens, N., Dehaene, W., & Verbauwhede, I. (2015, March). Embedded HW/SW platform for on-the-fly testing of true random number generators. In 2015 Design, Automation & Test in Europe Conference & Exhibition (DATE) (pp. 345-350). IEEE.

Yang, Z. L., Guo, X. Q., Chen, Z. M., Huang, Y. F., & Zhang, Y. J. (2018). RNN-stega: Linguistic steganography based on recurrent neural networks. IEEE Transactions on Information Forensics and Security, 14(5), 1280-1295.

Yi, S., & Zhou, Y. (2018). Separable and reversible data hiding in encrypted images using parametric binary tree labeling. IEEE Transactions on Multimedia, 21(1),

51-64.

Yıldız, Ç., Ceritli, T. Y., Kurt, B., Sankur, B., & Cemgil, A. T. (2016, May). Attack detection in VOIP networks using Bayesian multiple change-point models. In 2016 24th Signal Processing and Communication Application Conference (SIU) (pp. 1301-1304). IEEE.

Yu, Z., Thomborson, C., Wang, C., Fu, J., & Wang, J. (2009, November). A security model for VoIP steganography. In 2009 International Conference on Multimedia Information Networking and Security (Vol. 1, pp. 35-40). IEEE.

Zander, S., Armitage, G., & Branch, P. (2007). Covert channels and countermeasures in computer network protocols [reprinted from ieee communications surveys and tutorials]. IEEE Communications Magazine, 45(12), 136-142.

Zar, J. (2005). VoIP security and privacy threat taxonomy. VOIPSA, 24 October 2005.

Zhang, X., Peng, F., & Long, M. (2018). Robust coverless image steganography based on DCT and LDA topic classification. IEEE Transactions on Multimedia, 20(12), 3223-3238.

Zhou, H., Chen, K., Zhang, W., Yao, Y., & Yu, N. (2018). Distortion Design for Secure Adaptive 3-D Mesh Steganography. IEEE Transactions on Multimedia, 21(6), 1384-1398.